

(19)



JAPANESE PATENT OFFICE

## PATENT ABSTRACTS OF JAPAN

(11) Publication number: **10066082 A**(43) Date of publication of application: **06.03.98**

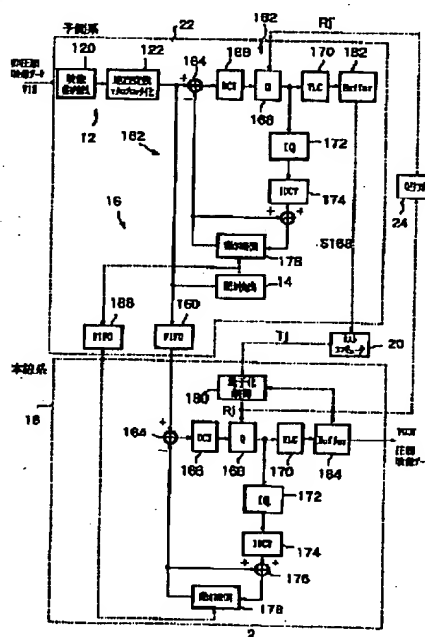
(51) Int. Cl.

**H04N 7/32**(21) Application number: **08216481**(71) Applicant: **SONY CORP**(22) Date of filing: **16.08.96**(72) Inventor: **KITAMURA TAKUYA****(54) VIDEO DATA COMPRESSOR AND ITS METHOD****(57) Abstract:**

**PROBLEM TO BE SOLVED:** To apply compression coding to non-compressed video data by adjusting a compression factor based on an estimated compressed video data amount.

**SOLUTION:** A quantization value generating section 24 calculates a mean value of a quantization value  $R_n$  of a quantization control circuit 180 while an encoder 18 applies compression coding to an  $n$ -th picture of non-compression video data VIN to generate a quantization value  $R_n'$ , and it is set to a quantization circuit 168 of an encoder 162 as a quantization value  $R_{n+L+1}'$  used when the encoder 162 applies compression coding to an  $(n+L+1)$ th picture ( $L$  is number of pictures delayed by an FIFO memory 160) of the non-compression video data VIN. A host computer 20 generates an object data amount  $T_j$  based on the amount of the compression video data generated by the encoder 162 and the encoder 18 conducts compression code processing so that the amount of the compression video data approaches the object data amount  $T_j$ .

COPYRIGHT: (C)1998,JPO



(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平10-66082

(43) 公開日 平成10年(1998) 3月6日

(51) Int.Cl.<sup>6</sup>

H 0 4 N 7/32

識別記号

庁内整理番号

F I

H 0 4 N 7/137

技術表示箇所

Z

審査請求 未請求 請求項の数12 O L (全 29 頁)

(21) 出願番号

特願平8-216481

(22) 出願日

平成8年(1996) 8月16日

(71) 出願人 000002185

ソニー株式会社

東京都品川区北品川6丁目7番35号

(72) 発明者 北村 卓也

東京都品川区北品川6丁目7番35号 ソニ

ー株式会社内

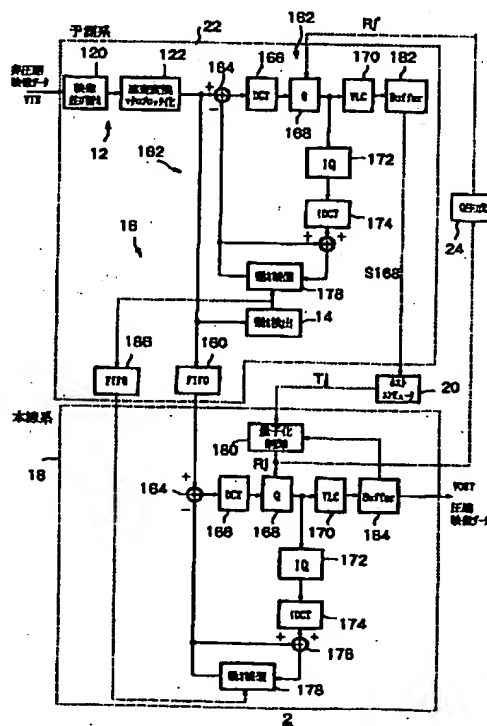
(74) 代理人 弁理士 佐藤 隆久

(54) 【発明の名称】 映像データ圧縮装置およびその方法

(57) 【要約】

【課題】 見積もった圧縮映像データのデータ量に基づいて圧縮率を調整して非圧縮映像データを圧縮符号化する。

【解決手段】 量子化値生成部24は、エンコーダ18が非圧縮映像データVINのn番目のピクチャーを圧縮符号化する間に、量子化制御回路180の量子化値R<sub>0</sub>の平均値を算出して量子化値R<sub>0</sub>'を生成し、エンコーダ162が非圧縮映像データVINの第n+L+1番目(LはFIFOメモリ160が遅延するピクチャーの数)のピクチャーを圧縮符号化する際の量子化値R<sub>n+L+1</sub>'として、エンコーダ162の量子化回路168に設定する。ホストコンピュータ20は、エンコーダ162が生成した圧縮映像データのデータ量に基づいて目標データ量T<sub>j</sub>を生成し、エンコーダ18は、圧縮映像データのデータ量が、目標データ量T<sub>j</sub>が示す値に近づくように圧縮符号化処理を行う。



## 【特許請求の範囲】

【請求項1】入力される映像データを、量子化処理を行う所定の圧縮方法により、外部から設定される第1の量子化値を用いてピクチャーごとに圧縮し、第1の圧縮映像データを生成する第1の圧縮手段と、

前記映像データが所定の数のピクチャー入力される時間だけ、前記映像データを遅延する遅延手段と、

前記映像データが所定の数のピクチャー入力される間に生成した前記第1の圧縮映像データのデータ量に基づいて、前記映像データを圧縮した後の圧縮映像データのデータ量の目標値をピクチャーごとに示す目標値データを生成する目標値データ生成手段と、

遅延した前記映像データを、前記第1の圧縮手段の前記所定の圧縮方法に対応し、量子化処理を行う所定の圧縮方法により、生成した前記目標値データが示すデータ量になるようにピクチャーごとに圧縮し、第2の圧縮映像データを生成する第2の圧縮手段と、

前記第2の圧縮手段の圧縮処理の内容に応じて、前記第1の量子化値を生成し、前記第1の圧縮手段に設定する第1の量子化値設定手段とを有する映像データ圧縮装置。

【請求項2】前記第1の圧縮手段の前記所定の圧縮方法は、所定の処理を前記映像データに対して行い、前記第1の量子化値を用いて量子化し、複数の種類のピクチャーを所定の順番で含むように圧縮する第1の圧縮方法であり、

前記第2の圧縮手段の前記所定の圧縮方法は、前記第1の圧縮方法と同じ所定の処理を前記映像データに対して行い、前記第2の量子化値を用いて量子化し、前記第1の圧縮方法と同じ複数の種類のピクチャーを、前記第1の圧縮方法と同じ順番で含むように圧縮する第2の圧縮方法である請求項1に記載の映像データ圧縮装置。

【請求項3】前記第1の圧縮手段は、

外部からピクチャーごとに設定される前記第1の量子化値を用いて、前記映像データを量子化する第1の量子化手段を有し、

前記第2の圧縮手段は、

外部から設定される第2の量子化値を用いて、遅延した前記映像データを量子化する第2の量子化手段と、

量子化した前記映像データのデータ量に基づいて、ピクチャーごとに、前記第2の圧縮映像データのデータ量が、前記目標値データが示すデータ量になるように前記第2の量子化値の値を調節して前記第2の量子化値を順次、生成し、前記第2の量子化手段に設定する第2の量子化値設定手段とを有する請求項2に記載の映像データ圧縮装置。

【請求項4】前記第1の量子化値設定手段は、生成した前記第2の量子化値に基づいて、前記第1の量子化値をピクチャーごとに生成する請求項3に記載の映像データ圧縮装置。

【請求項5】前記第1の量子化値設定手段は、生成した前記第2の量子化値のピクチャーごとの平均値に基づいて、前記第2の量子化値を生成する請求項3に記載の映像データ圧縮装置。

【請求項6】前記第1の量子化値設定手段は、第2の圧縮映像データのピクチャーの種類ごとに分けて前記第1の量子化値を生成し、前記第1の量子化値のピクチャーの種類と、前記第1の圧縮手段が生成する第1の圧縮映像データのピクチャーの種類とを合わせて、前記第1の量子化値を前記第1の量子化手段に設定する請求項3に記載の映像データ圧縮装置。

【請求項7】入力される映像データを、量子化処理を行う所定の圧縮方法により、外部から設定される第1の量子化値を用いてピクチャーごとに圧縮し、第1の圧縮映像データを生成し、

前記映像データが所定の数のピクチャー入力される時間だけ、前記映像データを遅延し、

前記映像データが所定の数のピクチャー入力される間に生成した前記第1の圧縮映像データのデータ量に基づいて、前記映像データを圧縮した後の圧縮映像データのデータ量の目標値をピクチャーごとに示す目標値データを生成し、

遅延した前記映像データを、前記所定の圧縮方法に対応し、量子化処理を行う所定の圧縮方法により、生成した前記目標値データが示すデータ量になるようにピクチャーごとに圧縮し、第2の圧縮映像データを生成し、

前記第2の圧縮映像データを生成する際の処理内容に応じて、前記第1の量子化値を生成する映像データ圧縮方法。

【請求項8】前記第1の圧縮映像データを生成する前記所定の圧縮方法は、所定の処理を前記映像データに対して行い、前記第1の量子化値を用いて量子化し、複数の種類のピクチャーを所定の順番で含むように圧縮する第1の圧縮方法であり、

前記第2の圧縮映像データを生成する前記所定の圧縮方法は、前記第1の圧縮方法と同じ所定の処理を前記映像データに対して行い、前記第2の量子化値を用いて量子化し、前記第1の圧縮方法と同じ複数の種類のピクチャーを、前記第1の圧縮方法と同じ順番で含むように圧縮する方法である請求項7に記載の映像データ圧縮方法。

【請求項9】前記第1の圧縮方法は、ピクチャーごとに設定される前記第1の量子化値を用いて前記映像データを量子化し、

前記第2の圧縮方法は、

順次、設定される第2の量子化値を用いて、遅延した前記映像データを量子化し、

量子化した前記映像データのデータ量に基づいて、ピクチャーごとに、前記第2の圧縮映像データのデータ量が、前記目標値データが示すデータ量になるように前記第2の量子化値の値を調節して前記第2の量子化値を生

成し、順次、設定する請求項8に記載の映像データ圧縮方法。

【請求項10】生成した前記第2の量子化値に基づいて、前記第1の量子化値をピクチャーごとに生成する請求項9に記載の映像データ圧縮方法。

【請求項11】生成した前記第2の量子化値のピクチャーごとの平均値に基づいて、前記第2の量子化値を生成する請求項9に記載の映像データ圧縮方法。

【請求項12】第2の圧縮映像データのピクチャーの種類ごとに分けて前記第1の量子化値を生成し、前記第1の量子化値のピクチャーの種類と、生成する第1の圧縮映像データのピクチャーの種類とを合わせて、前記第1の量子化値を設定する請求項9に記載の映像データ圧縮方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、非圧縮映像データを圧縮符号化する映像データ圧縮装置およびその方法に関する。

【0002】

【従来の技術および発明が解決しようとする課題】非圧縮のデジタル映像データをMPEG(moving picture experts group)等の方法により圧縮符号化して光磁気ディスク(MOディスク;magneto-optical disc)等の記録媒体に記録する際には、圧縮符号化後の圧縮映像データのデータ量(ビット量)を、伸長復号後の映像の品質を可能な限り高くしつつ、記録媒体の記録容量以下にする必要がある。この必要を満たすために、まず、非圧縮映像データを予備的に圧縮符号化して圧縮符号化後のデータ量を見積もり(1パス目)、次に、見積もったデータ量に基づいて圧縮率を調節し、圧縮符号化後のデータ量が記録媒体の記録容量以下になるように圧縮符号化する(2パス目)方法が採られる(以下、このような圧縮符号化方法を「2パスエンコード」とも記す)。

【0003】しかしながら、2パスエンコードにより圧縮符号化を行うと、同じ非圧縮映像データに対して同様な圧縮符号化処理を2回施す必要があり、時間がかかってしまう。また、1回の圧縮符号化処理で最終的な圧縮映像データを生成することができないために、撮影した映像データをそのまま実時間的(リアルタイム)に圧縮符号化し、記録することができない。

【0004】本発明は上述した従来技術の問題点に鑑みてなされたものであり、2パスエンコードによらずに所定のデータ量以下に圧縮符号化した圧縮映像データを伸長復号した場合に、高品質な映像を得ることができる映像データ圧縮装置およびその方法を提供することを目的とする。

【0005】また、本発明は、2パスエンコードによらずに、圧縮符号化後の圧縮映像データのデータ量を見積もり、見積もった圧縮映像データのデータ量に基づいて

圧縮率を調節して非圧縮映像データを圧縮符号化することができる映像データ圧縮装置およびその方法を提供することを目的とする。また、本発明は、2パスエンコードによらずに、高精度に圧縮符号化後の圧縮映像データのデータ量を見積もりながら、非圧縮映像データを圧縮符号化することができる映像データ圧縮装置およびその方法を提供することを目的とする。

【0006】

【課題を解決するための手段】上記目的を達成するために、本発明に係る映像データ圧縮装置は、入力される映像データを、量子化処理を行う所定の圧縮方法により、外部から設定される第1の量子化値を用いてピクチャーごとに圧縮し、第1の圧縮映像データを生成する第1の圧縮手段と、前記映像データが所定の数のピクチャー入力される時間だけ、前記映像データを遅延する遅延手段と、前記映像データが所定の数のピクチャー入力される間に生成した前記第1の圧縮映像データのデータ量に基づいて、前記映像データを圧縮した後の圧縮映像データのデータ量の目標値をピクチャーごとに示す目標値データを生成する目標値データ生成手段と、遅延した前記映像データを、前記第1の圧縮手段の前記所定の圧縮方法に対応し、量子化処理を行う所定の圧縮方法により、生成した前記目標値データが示すデータ量になるようにピクチャーごとに圧縮し、第2の圧縮映像データを生成する第2の圧縮手段と、前記第2の圧縮手段の圧縮処理の内容に応じて、前記第1の量子化値を生成し、前記第1の圧縮手段に設定する第1の量子化値設定手段とを有する。

【0007】好適には、前記第1の圧縮手段の前記所定の圧縮方法は、所定の処理を前記映像データに対して行い、前記第1の量子化値を用いて量子化し、複数の種類のピクチャーを所定の順番で含むように圧縮する第1の圧縮方法であり、前記第2の圧縮手段の前記所定の圧縮方法は、前記第1の圧縮方法と同じ所定の処理を前記映像データに対して行い、前記第2の量子化値を用いて量子化し、前記第1の圧縮方法と同じ複数の種類のピクチャーを、前記第1の圧縮方法と同じ順番で含むように圧縮する第2の圧縮方法である。

【0008】好適には、前記第1の圧縮手段は、外部からピクチャーごとに設定される前記第1の量子化値を用いて、前記映像データを量子化する第1の量子化手段を有し、前記第2の圧縮手段は、外部から設定される第2の量子化値を用いて、遅延した前記映像データを量子化する第2の量子化手段と、量子化した前記映像データのデータ量に基づいて、ピクチャーごとに、前記第2の圧縮映像データのデータ量が、前記目標値データが示すデータ量になるように前記第2の量子化値の値を調節して前記第2の量子化値を順次、生成し、前記第2の量子化手段に設定する第2の量子化値設定手段とを有する。

【0009】好適には、前記第1の量子化値設定手段

は、生成した前記第2の量子化値に基づいて、前記第1の量子化値をピクチャーごとに生成する。

【0010】好適には、前記第1の量子化値設定手段は、生成した前記第2の量子化値のピクチャーごとの平均値に基づいて、前記第2の量子化値を生成する。

【0011】好適には、前記第1の量子化値設定手段は、第2の圧縮映像データのピクチャーの種類ごとに分けて前記第1の量子化値を生成し、前記第1の量子化値のピクチャーの種類と、前記第1の圧縮手段が生成する第1の圧縮映像データのピクチャーの種類とを合わせて、前記第1の量子化値を前記第1の量子化手段に設定する。

【0012】本発明に係る映像データ圧縮装置において、第1の圧縮手段は、外部に接続された編集装置あるいはビデオテープレコーダ（VTR）装置等の映像処理装置から入力される非圧縮映像データを、第1の圧縮方法により、ピクチャーごとに圧縮符号化する（以下、「圧縮符号化」を、単に「圧縮」とも記す）。第1の圧縮手段としては、例えば、離散的コサイン変換（DCT）処理し、第2の圧縮手段からフィードバック制御される第1の量子化値で量子化し、さらに可変長符号化して、圧縮後の映像データが、複数の種類のピクチャー

〔I（intra coded）フレーム、B（bidirectionally predictive coded）ピクチャーおよびP（predictive coded）ピクチャー〕の組み合わせを所定の順番（ピクチャータイプシーケンス）で含むように圧縮符号化するMPEG（moving picture experts group）方式が用いられる。なお、後述するように、第1の圧縮手段が生成した第1の圧縮映像データのデータ量は、第2の圧縮手段が生成する圧縮映像データ（第2の圧縮映像データ）のデータ量を予測するために用いられる。

【0013】第1の圧縮手段が、第2の圧縮映像データのデータ量の予測のために必要な枚数のピクチャーの圧縮映像データ（第1の圧縮映像データ）を生成することができるよう、遅延手段は、外部から入力される非圧縮映像データを、所定数のピクチャーの分だけ遅延させ、第2の圧縮手段に供給する。遅延手段が、非圧縮映像データを上記所定数のピクチャーの分だけ遅延している間に、目標量データ生成手段は、生成された第1の圧縮映像データのデータ量に基づいて、第2の圧縮映像データの目標値をピクチャーごとに示す目標値データを生成する。なお、既に生成された第1の圧縮映像データのデータ量に基づいて、未生成の第1の圧縮映像データのデータ量を算出し、既に生成された第1の圧縮映像データのデータ量と、算出した未生成の圧縮映像データのデータ量とに基づいて、第2の圧縮映像データのデータ量の目標値を生成するように目標量データ生成手段を構成することも可能である。

【0014】第2の圧縮手段は、外部から入力される非

圧縮映像データを、第2の圧縮方法によりピクチャーごとに圧縮する。第2の圧縮方法としては、例えば、第1の圧縮方法と同様に、外部から入力される映像データを、DCT処理し、第2の圧縮映像データのピクチャーそれぞれのデータ量が、目標データ量が示すデータ量にほぼ等しくなるように、順次、値が調節される第1の量子化値で量子化し、さらに可変長符号化して、圧縮後の映像データ（第2の圧縮映像データ）のピクチャータイプシーケンスが、第1の圧縮方法と同じになるように圧縮符号化するMPEG方式が用いられる。

【0015】第1の量子化値設定手段は、例えば、第2の圧縮手段が、量子化処理段階で用いる第2の量子化値の平均値を算出して第1の量子化値を生成し、第1の圧縮手段にフィードバックする。このように、第1の量子化値のフィードバックを行うことにより、第1の圧縮手段が量子化処理に用いる第1の量子化値、および、目標量データ生成手段が生成する第2の圧縮映像データのデータ量の目標値が最適化され、この結果、第2の圧縮手段が用いる第2の量子化値も最適化され、第2の圧縮映像データの品質が向上する。

【0016】また、本発明に係る映像データ圧縮方法は、入力される映像データを、量子化処理を行う所定の圧縮方法により、外部から設定される第1の量子化値を用いてピクチャーごとに圧縮し、第1の圧縮映像データを生成し、前記映像データが所定の数のピクチャー入力される時間だけ、前記映像データを遅延し、前記映像データが所定の数のピクチャー入力される間に生成した前記第1の圧縮映像データのデータ量に基づいて、前記映像データを圧縮した後の圧縮映像データのデータ量の目標値をピクチャーごとに示す目標値データを生成し、遅延した前記映像データを、前記所定の圧縮方法に対応し、量子化処理を行う所定の圧縮方法により、生成した前記目標値データが示すデータ量になるようにピクチャーごとに圧縮し、第2の圧縮映像データを生成し、前記第2の圧縮映像データを生成する際の処理内容に応じて、前記第1の量子化値を生成する。

【0017】

【発明の実施の形態】

#### 第1実施形態

以下、本発明の第1の実施形態を説明する。MPEG方式といった映像データの圧縮符号化方式により、高い周波数成分が多い絵柄、あるいは、動きが多い絵柄といった難度（difficulty）が高い映像データを圧縮符号化すると、一般的に圧縮に伴う歪みが生じやすくなる。このため、難度が高い映像データは低い圧縮率で圧縮符号化する必要があり、難度が高いデータを圧縮符号化して得られる圧縮映像データに対しては、難度が低い絵柄の映像データの圧縮映像データに比べて、多くの目標データ量を配分する必要がある。

【0018】このように、映像データの難度に対して適

応的に目標データ量を配分するためには、従来技術として示した2パスエンコード方式が有効である。しかしながら、2パスエンコード方式は、実時間的な圧縮符号化に不向きである。第1の実施形態として示す簡易2パスエンコード方式は、かかる2パスエンコード方式の問題点を解決するためになされたものであり、非圧縮映像データを予備的に圧縮符号化して得られる圧縮映像データの難度データから非圧縮映像データの難度を算出し、予備的な圧縮符号化により算出した難度に基づいて、FIFOメモリ等により所定の時間だけ遅延した非圧縮映像データの圧縮率を適応的に制御することができる。

【0019】図1は、本発明に係る映像データ圧縮装置1の構成を示す図である。図1に示すように、映像データ圧縮装置1は、圧縮符号化部10およびホストコンピュータ20から構成され、圧縮符号化部10は、エンコーダ制御部12、動き検出器(motion estimator)14、簡易2パス処理部16、第2のエンコーダ(encoder)18から構成され、簡易2パス処理部16は、FIFOメモリ160および第1のエンコーダ162から構成される。なお、簡易2パス処理部16は、圧縮映像データのデータ量を予測するために用いられることから予測系とも呼ばれ、エンコーダ18は、最終的な圧縮映像データを生成することから、本線系とも呼ばれる。映像データ圧縮装置1は、これらの構成部分により、編集装置およびビデオテープレコーダ装置等の外部機器(図示せず)から入力される非圧縮映像データVINに対して、上述した簡易2パスエンコードを実現する。

【0020】映像データ圧縮装置1において、ホストコンピュータ20は、映像データ圧縮装置1の各構成部分の動作を制御する。また、ホストコンピュータ20は、簡易2パス処理部16のエンコーダ162が非圧縮映像データVINを予備的に圧縮符号化して生成した圧縮映像データのデータ量、DCT処理後の映像データの直流成分(DC成分)の値および直流成分(AC成分)の電力値を制御信号C16を介して受け、受けたこれらの値に基づいて圧縮映像データの絵柄の難度を算出する。さらに、ホストコンピュータ20は、算出した難度に基づいて、エンコーダ18が生成する圧縮映像データの目標データ量 $T_1$ を制御信号C18を介してピクチャーごとに割り当て、エンコーダ18の量子化回路166(図3)に設定し、エンコーダ18の圧縮率をピクチャー単位に適応的に制御する。

【0021】さらに具体例として、エンコーダ162が生成する圧縮映像データのデータ量を、そのまま難度データとして用いる場合を例に、ホストコンピュータ20の目標データ量 $T_1$ の算出動作を説明する。例えば、簡易2パス処理部16のFIFOメモリ160が、非圧縮映像データVINに対して、映像データ15ピクチャー分の時間遅延を与え、エンコーダ18が6Mbpsの圧縮映像データを生成する場合には、FIFOメモリ16

0にバッファリングされている15ピクチャー分の映像データを、エンコーダ18が圧縮符号化した後の圧縮映像データには、3Mビットのデータ量が割り当てられることになる。

【0022】また、例えば、エンコーダ162が、固定の量子化値(図3を参照して後述する)を用いて、下の表1に示す通りのデータ量の圧縮映像データを生成した場合、表1に示した15ピクチャー分の圧縮映像データのデータ量の総和は、1744380ビットとなる。このような場合、ホストコンピュータ20は、エンコーダ18に設定される目標データ量 $T_1$ を、各ピクチャーのデータ量(表1)それぞれに、 $3000000/1744380$ を乗算して、下の表2に示す各ピクチャーの目標データ量 $T_1$ を算出する。

【0023】

【表1】

frame No.	符号量
0	92860
1	71560
2	260130
3	98530
4	77560
5	161960
6	75100
7	80800
8	165980
9	99190
10	71710
11	180260
12	92770
13	75310
14	140660

【0024】

【表2】

frame No.	符号量
0	159701
1	123070
2	447374
3	169453
4	133388
5	278540
6	129158
7	138961
8	285454
9	170588
10	123327
11	310013
12	159547
13	129510
14	241908

【0025】エンコーダ制御部12は、非圧縮映像データVINのピクチャーの有無をホストコンピュータ20に通知し、さらに、非圧縮映像データVINのピクチャーごとに圧縮符号化のための前処理を行う。つまり、エンコーダ制御部12は、入力された非圧縮映像データを符号化順に並べ替え、ピクチャー・フィールド変換を行い、非圧縮映像データVINが映画の映像データである場合に3:2プルダウン処理（映画の24フレーム/秒の映像データを、30フレーム/秒の映像データに変換し、冗長性を圧縮符号化前に取り除く処理）等を行い、映像データS12として簡易2パス処理部16のFIFOメモリ160およびエンコーダ162に対して出力する。動き検出器14は、非圧縮映像データの動きベクトルの検出を行い、エンコーダ制御部12およびエンコーダ162、18に対して出力する。

【0026】簡易2パス処理部16において、FIFOメモリ160は、エンコーダ制御部12から入力された映像データS12を、例えば、非圧縮映像データVINが、L（Lは整数）ピクチャー入力される時間だけ遅延し、遅延映像データS16としてエンコーダ18に対して出力する。

【0027】図2は、図1に示した簡易2パス処理部16のエンコーダ162の構成を示す図である。エンコーダ162は、例えば、図2に示すように、加算回路164、DCT回路166、量子化回路（Q）168、可変長符号化回路（VLC）170、逆量子化回路（IQ）172、逆DCT（IDCT）回路174、加算回路176および動き補償回路178から構成される一般的な映像データ用圧縮符号化器であって、入力される映像データS12をMPEG方式等により圧縮符号化して、所定のピクチャータイプシーケンスの圧縮映像データを生成し、生成した圧縮映像データのピクチャーごとのデータ量等をホストコンピュータ20に対して出力する。

【0028】加算回路164は、加算回路176の出力データを映像データS12から減算し、DCT回路166に対して出力する。DCT回路166は、加算回路164から入力される映像データを、例えば、16画素×16画素のマクロブロック単位に離散コサイン変換（DCT）処理し、時間領域のデータから周波数領域のデータに変換して量子化回路168に対して出力する。また、DCT回路166は、DCT後の映像データのDC成分の値およびAC成分の電力値をホストコンピュータ20に対して出力する。

【0029】量子化回路168は、DCT回路166から入力された周波数領域のデータを、固定の量子化値Qで量子化し、量子化データとして可変長符号化回路170および逆量子化回路172に対して出力する。可変長符号化回路170は、量子化回路168から入力された量子化データを可変長符号化し、可変長符号化の結果として得られた圧縮映像データのデータ量を、制御信号C

16を介してホストコンピュータ20に対して出力する。逆量子化回路172は、可変長符号化回路168から入力された量子化データを逆量子化し、逆量子化データとして逆DCT回路174に対して出力する。

【0030】逆DCT回路174は、逆量子化回路172から入力される逆量子化データに対して逆DCT処理を行い、加算回路176に対して出力する。加算回路176は、動き補償回路178の出力データおよび逆DCT回路174の出力データを加算し、加算回路164および動き補償回路178に対して出力する。動き補償回路178は、加算回路176の出力データに対して、動き検出器14から入力される動きベクトルに基づいて動き補償処理を行い、加算回路176に対して出力する。

【0031】図3は、図1に示したエンコーダ18の構成を示す図である。図3に示すように、エンコーダ18は、図2に示したエンコーダ162に、量子化制御回路180を加えた構成になっている。エンコーダ18は、これらの構成部分により、ホストコンピュータ20から設定される目標データ量 $T_j$ に基づいて、FIFOメモリ160によりLピクチャー分遅延された遅延映像データS16に対して動き補償処理、DCT処理、量子化処理および可変長符号化処理を施して、MPEG方式等のエンコーダ162と同じピクチャータイプシーケンスの圧縮映像データVOU'Tを生成し、外部機器（図示せず）に出力する。

【0032】エンコーダ18において、量子化制御回路180は、可変長符号化回路170が出力する圧縮映像データVOU'Tのデータ量を順次、監視し、遅延映像データS16の第j番目のピクチャーから最終的に生成される圧縮映像データのデータ量が、ホストコンピュータ20から設定された目標データ量 $T_j$ に近づくように、順次、量子化回路168に設定する量子化値 $R_j$ を調節する。また、可変長量子化回路170は、圧縮映像データVOU'Tを外部に出力する他に、遅延映像データS16を圧縮符号化して得られた圧縮映像データVOU'Tの実際のデータ量 $S_j$ を制御信号C18を介してホストコンピュータ20に対して出力する。

【0033】以下、第1の実施形態における映像データ圧縮装置1の簡易2パスエンコード動作を説明する。図4（A）～（C）は、第1の実施形態における映像データ圧縮装置1の簡易2パスエンコードの動作を示す図である。エンコーダ制御部12は、映像データ圧縮装置1に入力された非圧縮映像データVINに対して、エンコーダ制御部12により符号化順にピクチャーを並べ替える等の前処理を行い、図4（A）に示すように映像データS12としてFIFOメモリ160およびエンコーダ162に対して出力する。なお、エンコーダ制御部12によるピクチャーの順番並べ替えにより、図4等に示すピクチャーの符号化の順番と伸長復号後の表示の順番とは異なる。

【0034】FIFOメモリ160は、入力された映像データS12の各ピクチャーをLピクチャー分だけ遅延し、エンコーダ18に対して出力する。エンコーダ162は、入力された映像データS12のピクチャーを予備的に順次、圧縮符号化し、第J（Jは整数）番目のピクチャーを圧縮符号化して得られた圧縮符号化データのデータ量、DCT処理後の映像データのDC成分の値、および、AC成分の電力値をホストコンピュータ20に対して出力する。

【0035】例えば、エンコーダ18に入力される遅延映像データS16は、FIFOメモリ160によりLピクチャーだけ遅延されているので、図4（B）に示すように、エンコーダ18が、遅延映像データS16の第J（Jは整数）番目のピクチャー（図4（B）のピクチャーa）を圧縮符号化している際には、エンコーダ162は、映像データS12の第J番目のピクチャーからLピクチャー分先の第（J+L）番目のピクチャー（図4

$$T_j = R'_j \times \frac{D_j}{\sum_{k=j}^{j+L-1} D_k} \quad (1)$$

【0039】但し、式1において、 $D_j$ は映像データS12の第J番目のピクチャーの実難度データであり、 $R'_j$ は、映像データS12、S16の第J番目～第（J+L-1）番目のピクチャーに割り当てることができる目標データ量の平均であり、 $R'_j$ の初期値（ $R'_1$ ）は、圧縮映像データの各ピクチャーに平均して割り

$$R'_1 = (\text{Bit\_rate} \times L) \times \text{Picture\_rate} \quad (2)$$

【0041】

$$R'_{j+1} = R'_j - S_j + F_{j+L} \quad (3)$$

【0042】なお、式3中の数値ビットレート（Bit rate）は、通信回線の伝送容量や、記録媒体の記録容量に基づいて決められる1秒当たりのデータ量（ビット量）を示し、ピクチャーレート（Picture rate）は、映像データに含まれる1秒当たりのピクチャーの数（30枚/秒（NTSC）、25枚/秒（PAL））を示し、数値 $F_{j+L}$ は、ピクチャータイプに応じて定められるピクチャー当たりの平均データ量を示す。エンコーダ18のDCT回路166は、入力される遅延映像データS16の第J番目のピクチャーをDCT処理し、量子化回路168に対して出力する。量子化回路168は、DCT回路166から入力された第J番目のピクチャーの周波数領域のデータを、量子化制御回路180が目標データ量 $T_j$ に基づいて調節する量子化値 $R_j$ により量子化し、量子化データとして可変長符号化回路170に対して出力する。可変長符号化回路170は、量子化回路168から入力された第J番目のピクチャーの量子化データを可変長符号化して、ほぼ、目標データ量 $T_j$ に近いデータ量の圧縮映像データVOUを生成して出力する。

【0043】同様に、図4（B）に示すように、エンコーダ18が、遅延映像データS16の第（J+1）番目

（B）のピクチャーb）を圧縮符号化していることになる。

【0036】従って、エンコーダ18が遅延映像データS16の第J番目のピクチャーの圧縮符号化を開始する際には、エンコーダ162は映像データS12の第J番目～第（J+L-1）番目のピクチャー（図4（B）の範囲c）の圧縮符号化を完了しており、これらのピクチャーの圧縮符号化後の実難度データ $D_j, D_{j+1}, D_{j+2}, \dots, D_{j+L-1}$ は、ホストコンピュータ20により既に算出されている。

【0037】ホストコンピュータ20は、下に示す式1により、エンコーダ18が遅延映像データS16の第J番目のピクチャーを圧縮符号化して得られる圧縮映像データに割り当てる目標データ量 $T_j$ を算出し、算出した目標データ量 $T_j$ を量子化制御回路180に設定する。

【0038】

【数1】

当て可能な目標データ量であり、下に示す式2で表され、エンコーダ18が圧縮映像データを1ピクチャー分生成する度に、式3に示すように更新される。

【0040】

【数2】

のピクチャー（図4（C）のピクチャーa'）を圧縮符号化している際には、エンコーダ162は、映像データS12の第（J+1）番目～第（J+L）番目のピクチャー（図4（C）の範囲c'）の圧縮符号化を完了し、これらのピクチャーの実難度データ $D_{j+1}, D_{j+2}, D_{j+3}, \dots, D_{j+L}$ は、ホストコンピュータ20により既に算出されている。

【0044】ホストコンピュータ20は、式1により、エンコーダ18が遅延映像データS16の第（J+1）番目のピクチャーを圧縮符号化して得られる圧縮映像データに割り当てる目標データ量 $T_{j+1}$ を算出し、エンコーダ18の量子化制御回路180に設定する。

【0045】エンコーダ18は、ホストコンピュータ20から量子化制御回路180に設定された目標データ量 $T_j$ に基づいて第（J+1）番目のピクチャーを圧縮符号化し、目標データ量 $T_{j+1}$ に近いデータ量の圧縮映像データVOUを生成して出力する。さらに以下、同様に、映像データ圧縮装置1は、遅延映像データS16の第k番目のピクチャーを、量子化値 $Q_k$ （ $k = j+2, j+3, \dots$ ）をピクチャーごとに変更して順次、圧縮符号化し、圧縮映像データVOUとして出力する。



【0046】以上説明したように、第1の実施形態に示した映像データ圧縮装置1によれば、短時間で非圧縮映像データVINの絵柄の難度を算出し、算出した難度に応じた圧縮率で適応的に非圧縮映像データVINを圧縮符号化することができる。つまり、第1の実施形態に示した映像データ圧縮装置1によれば、2パスエンコード方式と異なり、ほぼ実時間的に、非圧縮映像データVINの絵柄の難度に基づいて適応的に非圧縮映像データVINを圧縮符号化をすることができ、実況放送といった実時間性を要求される用途に応用可能である。なお、第1の実施形態に示した他、本発明に係るデータ多重化装置1は、エンコーダ162が圧縮符号化した圧縮映像データのデータ量を、そのまま難度データとして用い、ホストコンピュータ20の処理の簡略化を図る等、種々の構成を採ることができる。

#### 【0047】第2実施形態

第1の実施形態に示した簡易2パスエンコード方式によれば、実時間かつ、絵柄の難度に応じた適応的な非圧縮映像データに対する圧縮符号化処理が可能である。しかしながら、第1の実施形態に示した簡易2パスエンコード方式を用いた場合、実時間性が厳しく要求される場合には、FIFOメモリ160の遅延時間を大きくすることができず、真に適切な目標データ量 $T_j$ の算出が難しく、圧縮映像データVOUを伸長復号して得られる映像の品質が低下してしまう可能性がある。

【0048】第2の実施形態においては、第1の実施形態に示した映像データ圧縮装置1(図1)を用い、ホストコンピュータ20の処理内容を変更して、FIFOメモリ160の遅延時間を長くしなくても適切な目標データ量 $T_j$ の値を得ることができるよう、非圧縮映像データをLピクチャー分、予備的に圧縮符号化して得られた圧縮映像データの第j番目のピクチャー～第(j+L-1)番目のピクチャーの実難度データ $D_j \sim D_{j+L-1}$ から、圧縮映像データの第(j+L)番目のピクチャー～第(j+L+B)番目のピクチャー(Bは整数)の難度データ(予測難度データ) $D_{j+L} \sim D_{j+L+B}$ を算出し、実際に得られた難度データ $D_j \sim D_{j+L-1}$ (実難度データ)および予測によって得られた難度データ $D'_{j+L} \sim D'_{j+L+B}$ に基づいて、第1の実施形態に示した簡易2パスエンコード方式よりも適切な目標データ量 $T_j$ の値を得ることができる圧縮符号化方式(予測簡易2パスエンコード方式)を説明する。

【0049】まず、第2の実施形態で説明する予測簡易2パスエンコード方式を概念的に説明する。予測簡易2パスエンコード方式は、徐々に絵柄が難しくなっていく、つまり、徐々に圧縮符号化時のDCT処理後の高い周波数成分が多くなり、動きが速くなっていく非圧縮映像データの絵柄は、さらに難しくなっていく、逆に、徐々に絵柄が難しくなくなっていく(簡単になっていく)非圧縮映像データの絵柄は、さらに簡単になっていくと予測

すると予測可能であることを前提する。

【0050】つまり、予測簡易2パスエンコード方式は、ホストコンピュータ20が、この前提に基づいて、さらに絵柄が難しくなっていくと予測される場合には、さらに絵柄が難しいピクチャーに備えて、その時点で圧縮符号化しているピクチャーに割り当てる目標データ量を節約し、逆に、さらに絵柄が簡単になっていくと予測される場合には、その時点で圧縮符号化しているピクチャーに割り当てる目標データ量を増やすようにエンコーダ18に対する圧縮率の制御を行う。

【0051】さらに、予測簡易2パスエンコード方式の概念的な説明を続ける。映像データは、一般的に、時間方向および空間方向について相関性が高く、映像データの圧縮符号化は、これらの相関性に着目し、冗長性を除くことにより行われる。時間方向について相関性が高いということは、現時点の非圧縮映像データのピクチャーの難度とそれ以降の非圧縮映像データのピクチャーの難度とが近いということを意味する。また、難度の増減の傾向も、現時点までの難度の増減の傾向がそれ以降も続くことが多い。

【0052】具体例を挙げると、カメラが静止状態からゆっくりとカメラを水平方向に回し初め、最後に一定の回転速度で回転しながら、静止している物体を撮影する場合の非圧縮映像データの絵柄を考える。最初はカメラが停止状態であるため、静止映像が撮影され、絵柄の難度は低くなる。次に、カメラを回し始めて1～2秒後に一定の回転速度になると仮定すると、カメラを回し始めて1～2秒間は絵柄の難度は高くなる傾向を示す。この状態を、映像データ圧縮装置1側から見ると、数GOP分の圧縮映像データを生成する間、入力される非圧縮映像データの絵柄の難度が高くなる傾向が続くことになる。

【0053】従って、この具体例に示したような場合には、非圧縮映像データの絵柄の難度が増大傾向を示した場合に、それ以降の絵柄の難度が増大傾向を示すと予測するのは妥当である。以下に説明する予測簡易2パスエンコード方式は、このような難度および難度の増減傾向の時間的相関性を積極的に利用して、圧縮映像データの各ピクチャーに対して、第1の実施形態に示した簡易2パスエンコード方式においてよりも適切な目標データ量の割り当てを行おうとするものである。

【0054】以下、第2の実施形態における映像データ圧縮装置1の予測簡易2パスエンコードの動作を説明する。図5(A)～(C)は、第2の実施形態における映像データ圧縮装置1の予測簡易2パスエンコードの動作を示す図である。エンコーダ制御部12は、第1の実施形態においてと同様に、映像データ圧縮装置1に入力された非圧縮映像データVINに対して、エンコーダ制御部12により符号化順にピクチャーを並べ替える等の前処理を行い、図5(A)に示すように映像データS12

としてFIFOメモリ160およびエンコーダ162に対して出力する。

【0055】FIFOメモリ160は、第1の実施形態においてと同様に、入力された映像データS12の各ピクチャーをLピクチャー分だけ遅延し、エンコーダ18に対して出力する。エンコーダ162は、第1の実施形態においてと同様に、入力された映像データS12のピクチャーを予備的に順次、圧縮符号化し、第J（Jは整数）番目のピクチャーを圧縮符号化して得られた圧縮符号化データのデータ量、DCT処理後の映像データのDC成分の値およびAC成分の電力値をホストコンピュータ20に対して出力する。ホストコンピュータ20は、エンコーダ162から入力されたこれらの値に基づいて、実難度データ $D_j$ を順次、算出する。

【0056】例えば、エンコーダ18に入力される遅延映像データS16は、FIFOメモリ160によりLピクチャーだけ遅延されているので、図5（B）に示すように、エンコーダ18が、遅延映像データS16の第J番目のピクチャー（図5（B）のピクチャーa）を圧縮符号化している際には、エンコーダ162は、第1の実施形態においてと同様に、映像データS12の第J番目のピクチャーからLピクチャー分先の第（J+L）番目のピクチャー（図5（B）のピクチャーb）を圧縮符号化していることになる。

【0057】従って、エンコーダ18が遅延映像データS16の第J番目のピクチャーの圧縮符号化を開始する際には、エンコーダ162は映像データS12の第（J-A）番目～第（J+L-1）番目のピクチャー（図5

（B）の範囲c、但し、図5はA=0の場合を示す）の圧縮符号化を完了し、これらのピクチャーの圧縮符号化後のデータ量、および、DCT処理後の映像データのDC成分の値およびAC成分の電力値をホストコンピュータ20に対して出力している。ホストコンピュータ20は、エンコーダ162から入力されたこれらの値に基づいて、難度データ（実難度データ、図5（B）の範囲d） $D_{j-A}, D_{j-A+1}, \dots, D_j, D_{j+1}, D_{j+2}, \dots, D_{j+L-1}$ の算出を既に終了している。なお、Aは整数であり、正負を問わない。

【0058】ホストコンピュータ20は、実難度データ $D_{j-A}, D_{j-A+1}, \dots, D_j, D_{j+1}, D_{j+2}, \dots, D_{j+L-1}$ に基づいて、映像データS12の第（J+L）番目～第（J+L+B）番目のピクチャーの圧縮符号化後の難度データ（予測難度データ、図5（B）の範囲e） $D'_{j+L}, D'_{j+L+1}, D'_{j+L+2}, \dots, D'_{j+L+B}$ を予測し、下に示す式4により、遅延映像データS16の第J番目のピクチャーの圧縮符号化後の目標データ量 $T_j$ を算出する。従って、遅延映像データS16の第J番目のピクチャーの圧縮符号化後の目標データ量 $T_j$ を算出するために、実難度データと予測難度データとを含めて、図5（B）の範囲cの（A+L+B+1）ピクチャー分の難度データを用いることになる。なお、実難度データに基づいて予測難度データを算出する方法は、第5の実施形態以降に詳述する。

【0059】

【数4】

$$T_j = R'_j \times \frac{D_j}{\sum_{k=j}^{j+L-1} D_k + \sum_{k=j+L}^{j+L+B-1} D'_k} \quad (4)$$

【0060】なお、式4の各記号は、式1の各記号に同じである。エンコーダ18は、第1の実施形態と同様に、ホストコンピュータ20により量子化制御回路180に設定された目標データ量 $T_j$ に基づいて、目標データ量 $T_j$ に近いデータ量の圧縮映像データVOUUTを生成して出力する。さらに、ホストコンピュータ20は、図5（B）に示した動作と同様に、遅延映像データS16の第（J+1）番目のピクチャー（図5（C）のピクチャーa'）に対しても、映像データS12の第（J+L+1）番目のピクチャー（図5（C）のピクチャーb'）以前の図5（C）の範囲d'の実難度データ $D_{j-A+1}, D_{j-A+2}, \dots, D_j, D_{j+1}, D_{j+2}, \dots, D_{j+L}$ 、および、図5（C）の範囲e'に示す予測難度データ、 $D'_{j+L+1}, D'_{j+L+2}, D'_{j+L+3}, \dots, D'_{j+L+B+1}$ 、つまり、図5（C）の範囲c'に示す実難度データと予測難度データとに基づいて、遅延映像データS16の第（J+1）番目のピクチャーの圧縮符号化後の目標データ量 $T_{j+1}$ を算出する。エンコーダ18は、

ホストコンピュータ20が算出した目標データ量 $T_{j+1}$ に基づいて、遅延映像データS16の第（J+1）番目のピクチャーを圧縮符号化し、目標データ量 $T_{j+1}$ に近いデータ量の圧縮符号化データVOUUTを生成する。なお、以上の映像データ圧縮装置1の予測簡易2パスエンコード動作は、遅延映像データS16の第（J+1）番目のピクチャーに対しても同様である。

【0061】以下、図6を参照して、第2の実施形態における映像データ圧縮装置1の動作を整理して説明する。図6は、第2の実施形態における映像データ圧縮装置1（図1）の動作を示すフローチャートである。図6に示すように、ステップ102（S102）において、ホストコンピュータ20は、式1等に用いられる数値J、 $R'_j$ を、 $J = -(L-1)$ 、 $R'_j = (\text{Bit rate} \times (L+B)) / \text{Picture rate}$ として初期化する。

【0062】ステップ104（S104）において、ホストコンピュータ20は、数値Jが0より大きいかな否かを判断する。数値Jが0より大きい場合にはS106の

処理に進み、小さい場合にはS110の処理に進む。ステップ106(S106)において、エンコーダ162は、映像データS12の第(J+L)番目のピクチャーを圧縮符号化し、実難度データ $D_{JHL}$ を生成する。ステップ108(S108)において、ホストコンピュータ20は数値Jをインクリメントする( $J=J+1$ )。ステップ110(S110)において、ホストコンピュータ20は、遅延映像データS16に第J番目のピクチャーが存在するか否かを判断する。第J番目のピクチャーが存在する場合にはS112の処理に進み、存在しない場合には圧縮符号化処理を終了する。

【0063】ステップ112(S112)において、ホストコンピュータ20は、数値Jが数値Aよりも大きいのか否かを判断する。数値Jが数値Aよりも大きい場合にはS114の処理に進み、小さい場合にはS116の処理に進む。ステップ114(S114)において、ホストコンピュータ20は、実難度データ $D_{J-A} \sim D_{JHL-1}$ に基づいて、予測難度データ $D'_{JHL} \sim D'_{JHL+B}$ を算出する。ステップ116(S116)において、ホストコンピュータ20は実難度データ $D_J \sim D_{JHL-1}$ から、予測難度データ $D'_{JHL} \sim D'_{JHL+B}$ を算出する。

【0064】ステップ118(S118)において、ホストコンピュータ20は、式4を用いて目標データ量 $T_J$ を算出し、エンコーダ18の量子化制御回路180に設定する。さらに、エンコーダ18は、量子化制御回路180に設定された目標データ量 $T_J$ に基づいて遅延映像データS16の第J番目のピクチャーを圧縮符号化し、第J番目のピクチャーから実際に得られた圧縮映像データのデータ量 $S_J$ をホストコンピュータ20に対して出力する。ステップ120(S120)において、ホストコンピュータ20は、エンコーダ18からのデータ量 $S_J$ を記憶し、さらに、映像データS12の第(J+L)番目のピクチャーの実難度データ $D_{JHL}$ を出力する。

【0065】ステップ122(S122)において、エンコーダ18は、遅延映像データS16の第J番目を圧縮符号化して得られた圧縮映像データVOUを外部に出力する。ステップ124(S124)において、ホストコンピュータ20は、ピクチャータイプに応じて、式3中に用いられる数値 $F_{JHL}$ を算出する。ステップ126(S126)において、ホストコンピュータ20は、式3に示した演算( $R'_{JHL} = R'_J - S_J + F_{JHL}$ )を行う。

【0066】以上説明したように、第2の実施形態に示した映像データ圧縮装置1による予測簡易2パスエンコードによれば、短時間で非圧縮映像データVINの絵柄の難度を算出し、算出した難度に基づいて予測した難度をさらに用いて適応的に非圧縮映像データVINを圧縮符号化することができ、簡易2パスエンコード方式に比べて、より適切な目標データ量を圧縮映像データの各ピ

クチャーに割り当てることが可能である。従って、予測簡易2パスエンコード方式による圧縮映像データを伸長復号した場合、簡易2パスエンコード方式による圧縮映像データを伸長復号した場合に比べて、より高品質な映像を得ることができる。

### 【0067】第3実施形態

以下、本発明の第3の実施形態を説明する。エンコーダ162およびエンコーダ18(図1)は、MPEG方式等により、非圧縮映像データを、複数のIピクチャー(intra coded picture)、Bピクチャー(bi-directionally coded picture)およびPピクチャー(predictive coded picture)またはこれらの組み合わせから構成されるGOP(group of pictures)単位に圧縮符号化する。このため、第1の実施形態に示した簡易2パスエンコード方式も、GOPの枠内に区切って行うのが便利である。第3の実施形態においては、映像データ圧縮装置1(図1)を用い、GOPの枠内で行う簡易2パスエンコード方式を説明する。

【0068】図7(A)～(C)は、第3の実施形態において、本発明に係る映像データ圧縮装置1がGOPの枠内で行う簡易2パスエンコード方式を示す図である。なお、図7においては、最初のGOPの最初のピクチャーを1番目のピクチャーとして示している。

【0069】図7(A)に示すように、エンコーダ制御部12は、第1の実施形態と同様に、入力される非圧縮映像データVINに含まれるピクチャーの順番を並び替え、FIFOメモリ160およびエンコーダ162に対して出力する。FIFOメモリ160は、入力された映像データS12の各ピクチャーをNピクチャー分( $L=N$ , Nは整数)つまり、1GOP分だけ遅延し、エンコーダ18に対して出力する。エンコーダ162は、入力された映像データS12のピクチャーを予備的に順次、圧縮符号化し、第J(Jは整数)番目のピクチャーを圧縮符号化して得られた圧縮符号化データのデータ量、DCT後の映像データのDC成分の値およびAC成分の電力値を順次、ホストコンピュータ20に対して出力する。

【0070】例えば、エンコーダ18に入力される遅延映像データS16は、FIFOメモリ160によりNピクチャーだけ遅延されているので、図7(B)に示すように、エンコーダ18が、遅延映像データS16の第J番目のピクチャー(図7(B)のピクチャーa)を圧縮符号化している際には、エンコーダ162は、映像データS12の第J番目のピクチャーからNピクチャー分先の第(J+N)番目のピクチャー(図7(B)のピクチャーb)を圧縮符号化していることになる。

【0071】しかし、第3の実施形態においては、映像データ圧縮装置1はGOPの枠内で簡易2パスエンコードを行うので、ホストコンピュータ20は、エンコーダ18が遅延映像データS16の第J番目のピクチャーの

圧縮符号化を開始する際に、エンコーダ162が圧縮符号化を完了している映像データS12の第j番目のピクチャー～第(j+N-1)番目のピクチャーの内、第j番目～第N番目のピクチャー(図7(B)の範囲c)の圧縮符号化後の実難度データ(図7(B)の範囲d)D

$$T_j = R''_j \times \frac{D_j}{\sum_{k=j}^{j+N-1} D_k + \sum_{k=j+N}^{2N} D'_k}$$

(5)

【0073】ただし、式5中の $R''_j$ は、GOPに割り当てられた目標データ量から、遅延映像データS16の同一GOP内の第1番目のピクチャー～第(j-1)番目のピクチャーに対する圧縮符号化で既に実際に使用されたデータ量を減算した値、つまり、同一GOPの残り

$$R''_j = R'_{j-1} - S_j$$

【0075】従って、GOPの最初のピクチャーを圧縮符号化する場合の $R''_1$ (j=1)は、下に示す式7により、求められる。

$$R''_1 = G + R''_N$$

【0077】ただし、式7において、Gは1つのGOPに割り当てられる固定データ量( $G = (\text{Bit rate} \times N) / \text{picture rate}$ )であり、 $R''_N$ (j=N)の初期値は0である。

【0078】エンコーダ18は、第1の実施形態と同様に、量子化制御回路180にホストコンピュータ20から設定された目標データ量 $T_j$ に基づいて、データ量が目標データ量 $T_j$ に近い圧縮映像データVOUを生成して出力する。さらに、ホストコンピュータ20は、図7(B)に示した動作と同様に、遅延映像データS16の第(j+1)番目のピクチャー(図7(C)のピクチャーa')に対しても、映像データS12の第N番目のピクチャー以前の図7(C)の範囲d'の実難度データ $D_{j+1}, D_{j+2}, \dots, D_N$ 、つまり、図7(C)の範囲c'に示す難度データに基づいて、遅延映像データS16の第(j+1)番目のピクチャーの圧縮符号化後の目標データ量 $T_{j+1}$ を算出する。エンコーダ18は、ホストコンピュータ20が算出した目標データ量 $T_{j+1}$ に基づいて、遅延映像データS16の第(j+1)番目のピクチャーを圧縮符号化し、目標データ量 $T_{j+1}$ の圧縮符号化データVOUを生成する。なお、以上の映像データ圧縮装置1の簡易2パスエンコード動作は、遅延映像データS16の第(j+1)番目のピクチャーに対しても同様である。

【0079】以上説明したように、第3の実施形態に示した映像データ圧縮装置1による簡易2パスエンコードによれば、GOPの枠内で簡易2パスエンコードを行うことができるので、処理内容が簡単で、処理時間が短くてすむ。従って、実時間性を要求される実況中継の非圧縮映像データを圧縮符号化するために適しており、しかも、映像データ圧縮装置1の構成を簡単かつ安価に構成

$j, D_{j+1}, \dots, D_N$ を用いて目標データ量 $T_j$ を、下に示す式5により算出し、エンコーダ18の量子化制御回路180に設定する。

【0072】

【数5】

のピクチャーの圧縮映像データに割り当てられるべき目標データ量である。従って、 $R''_j$ は、下に示す式6により求められる。

【0074】

【数6】

(6)

【0076】

【数7】

(7)

することができる。

【0080】第4実施形態

以下、本発明の第4の実施形態を説明する。第4の実施形態においては、映像データ圧縮装置1(図1)によるGOPの枠内に区切って行う予測簡易2パスエンコード、つまり、同じエンコーダ18が遅延映像データS16の第j番目のピクチャーの圧縮符号化を行う際に、同一のGOPの各ピクチャーの実難度データ、および、次のGOPについて予測した予測難度データを用いて目標データ量 $T_j$ を算出し、エンコーダ18の量子化制御回路180に設定して圧縮符号化を行う方法を説明する。

【0081】図8(A)～(C)は、第4の実施形態において、本発明に係る映像データ圧縮装置1がGOPの枠内で行う予測簡易2パスエンコード方式を示す図である。なお、図8においては、GOPの最初のピクチャーを1番目のピクチャーとして示している。

【0082】図8(A)に示すように、エンコーダ制御部12は、第1の実施形態と同様に、入力される非圧縮映像データVINに含まれるピクチャーの順番を並び替え、FIFOメモリ160およびエンコーダ162に対して出力する。FIFOメモリ160は、入力された映像データS12の各ピクチャーをNピクチャー分つまり、1GOP分だけ遅延し、エンコーダ18に対して出力する。エンコーダ162は、入力された映像データS12のピクチャーを予備的に順次、圧縮符号化し、第j(jは整数)番目のピクチャーを圧縮符号化して得られた圧縮符号化データのデータ量、DCT後の映像データのDC成分の値およびAC成分の電力値を順次、ホストコンピュータ20に対して出力する。ホストコンピュータ20は、これらの値から実難度データ $D_j$ を算出する。

【0083】例えば、エンコーダ18に入力される遅延映像データS16は、FIFOメモリ160によりNピクチャーだけ遅延されているので、図8(B)に示すように、エンコーダ18が、遅延映像データS16の第j番目のピクチャー(図8(B)のピクチャーa)を圧縮符号化している際には、エンコーダ162は、映像データS12の第j番目のピクチャーからNピクチャー分先の第(j+N)番目のピクチャー(図8(B)のピクチャーb)を圧縮符号化していることになる。

【0084】しかし、第4の実施形態においては、映像データ圧縮装置1はGOPの枠内で簡易2パスエンコードを行うので、ホストコンピュータ20は、エンコーダ18が遅延映像データS16の第j番目のピクチャーの圧縮符号化を開始する際に、エンコーダ162が圧縮符

$$T_j = R''_j \times \frac{D_j}{\sum_{k=j}^{j+N-1} D_k + \sum_{k=j+N}^{2N} D'_k} \quad (8)$$

【0086】ただし、式8中において、j=1, 2, ..., Nであり、R''\_jは、2個のGOPに割り当てられた目標データ量から、2個のGOP内の遅延映像データS16の第1番目のピクチャー～第(j-1)番目のピクチャーに対する圧縮符号化で既に実際に使用されたデータ量を減算した値、つまり、2個のGOPの残りのピ

$$R''_j = R'_{j-1} - S_j \quad (9)$$

【0088】従って、図8の最初のGOPの最初のピクチャーを圧縮符号化する場合のR''\_1 (j=1)は、下に示す式10により、求められる。

$$R''_1 = G + R''_N$$

【0090】ただし、式10のR''\_Nの初期値はGである。

【0091】エンコーダ18は、第2の実施形態と同様に、量子化制御回路180にホストコンピュータ20から設定された目標データ量T\_jに基づいて、データ量が目標データ量T\_jに近い圧縮映像データVOU\_Tを生成して出力する。さらに、ホストコンピュータ20は、図8(B)に示した動作と同様に、遅延映像データS16の第(j+1)番目のピクチャー(図8(C)のピクチャーa')に対しても、映像データS12の第(j+N)番目のピクチャー以前の図8(C)の範囲d'の実難度データD\_{j+2}, D\_{j+3}, ..., D\_{N+1}つまり、図8

(C)の範囲c'に示す実難度データ、および、映像データS12の第(j+N+1)番目のピクチャー～第2N番目のピクチャーの予測難度データに基づいて、遅延映像データS16の第(j+1)番目のピクチャーの圧縮符号化後の目標データ量T\_{j+1}を算出する。

【0092】エンコーダ18は、ホストコンピュータ20が算出した目標データ量T\_{j+1}に基づいて、遅延映像データS16の第(j+1)番目のピクチャーを圧縮符

号化を完了している映像データS12の第j番目のピクチャー～第(j+N-1)番目のピクチャーの内、第j番目～第(j+N-1)番目のピクチャー(図8(B)の範囲c)の圧縮符号化後の難度データ(図8(B)の範囲d)を実難度データD\_j, D\_{j+1}, ..., D\_{j+N-1}として用い、さらに、同じGOPの第(j+1)番目のピクチャー～次のGOPの最後のピクチャー(第2N番目のピクチャー)までの予測難度データD'\_{j+N}, D'\_{j+N+1}, ..., D'\_{2N}を算出し、実難度データおよび予測難度データに基づいて目標データ量T\_jを、下に示す式8により算出し、エンコーダ18の量子化制御回路180に設定する。

【0085】

【数8】

クチャーの圧縮映像データに割り当てられるべき目標データ量である。従って、R''\_jは、下に示す式9により求められる。

【0087】

【数9】

【0089】

【数10】

(10)

号化し、目標データ量T\_{j+1}の圧縮符号化データVOU\_Tを生成する。なお、以上の映像データ圧縮装置1の簡易2パスエンコード動作は、遅延映像データS16の第(j+1)番目のピクチャーに対しても同様である。

【0093】以下、図9を参照して、第4の実施形態における映像データ圧縮装置1の動作を説明する。図9は、第4の実施形態における映像データ圧縮装置1の動作を示すフローチャートである。

【0094】図9に示すように、ステップ202(S202)において、ホストコンピュータ20は、数値GOP\_count, R\_Nを0クリアする(GOP\_count, R\_N=0)。ステップ204(S204)において、ホストコンピュータ20は、数値GOP\_countが0であるか否かを判断する。数値GOP\_countが0である場合には、ホストコンピュータ20はS206の処理に進み、0でない場合にはS210の処理に進む。

【0095】ステップ206(S206)において、映像データS12の第(j+N)番目のピクチャーを圧縮符号化し、結果として得られた圧縮映像データのデータ量、DCT処理後の映像データのDC成分の値およびA

C成分の電力値をホストコンピュータ20に対して出力する。ステップ208(S208)において、ホストコンピュータ20は、数値JがNであって、J=1である場合に、数値GOP<sub>count</sub>をインクリメントし、数値R''<sub>J</sub>をG+R''<sub>N</sub>(R''<sub>J</sub>=G+R''<sub>N</sub>;式7)とし、これ以外の場合には数値Jをインクリメントし、数値R''<sub>J</sub>をR''<sub>J-1</sub>-S<sub>J</sub>(R''<sub>J</sub>=R''<sub>J-1</sub>-S<sub>J</sub>;式6)とする。

【0096】ステップ210(S210)において、ホストコンピュータ20は、遅延映像データS16に第J番目のピクチャーが存在するか否かを判断する。第J番目のピクチャーが存在する場合にはS212の処理に進み、存在しない場合には圧縮符号化処理を終了する。ステップ212(S212)において、ホストコンピュータ20は、実難度データD<sub>J</sub>~D<sub>J+N-1</sub>から予測難度データD'<sub>J+N</sub>~D'<sub>2N</sub>を算出する。ステップ214(S214)において、ホストコンピュータ20は、実難度データD<sub>J</sub>~D<sub>J+N-1</sub>および予測難度データD'<sub>J+N</sub>~D'<sub>2N</sub>を用いて目標データ量T<sub>J</sub>を算出し、エンコーダ18の量子化制御回路180に設定する。エンコーダ18は、設定された目標データ量T<sub>J</sub>を用いて遅延映像データS16の第J番目のピクチャーを圧縮符号化する。

【0097】ステップ216(S216)において、エンコーダ18は、遅延映像データS16の第J番目のピクチャーを圧縮符号化して得られた圧縮映像データのデータ量S<sub>J</sub>をホストコンピュータ20に対して出力し、ホストコンピュータ20はデータ量S<sub>J</sub>を記憶する。エンコーダ162は、映像データS12の第(J+N)番目のピクチャーを圧縮符号化し、結果として得られた映像データのデータ量、DCT処理後の映像データのDC成分の値およびAC成分の電力値をホストコンピュータ20に対して出力する。ホストコンピュータ20は、これらの値に基づいて、実難度データD<sub>J+N</sub>を算出する。ステップ218(S218)において、エンコーダ18は、遅延映像データS16の第J番目の圧縮映像データVOUTを外部に出力する。

【0098】以上説明したように、第4の実施形態に示した映像データ圧縮装置1によるGOPの枠内での予測簡易2パスエンコードは、第3の実施形態に示したGOPの枠内での簡易2パスエンコードに比べ、より適切な目標データ量T<sub>J</sub>を算出することができるので、圧縮符号化の結果として得られる圧縮映像データVOUTを伸長復号した映像の品質を高めることができる。しかも、

$$D'' = \frac{\sum_{k=1}^{15} D_k}{15}$$

【0104】つぎに、ホストコンピュータ20は、下に示す式12により、15個の実難度データD<sub>1</sub>~D<sub>15</sub>の中心のデータ、つまり、実難度データD<sub>8</sub>が平均値D''

第2の実施形態に示したGOPの枠に納まらない予測簡易2パスエンコードに比べて処理内容が簡単で、処理時間が短くてすみ、実時間性を要求される実況中継の非圧縮映像データを圧縮符号化するために適しており、しかも、映像データ圧縮装置1の構成を簡単かつ安価に構成することができる。

#### 【0099】第5実施形態

以下、本発明の第5の実施形態として、実難度データに基づいて予測難度データを、単純な直線予測により算出する方法を説明する。なお、同一GOPに複数の種類のピクチャータイプが混在していると、難度データの予測が複雑化する。従って、説明の便宜上、第5の実施形態においては、GOPに含まれるピクチャーが全てIピクチャーであり、実難度データと予測難度データがそれぞれ15(L+A, B=15)である場合を例に説明する。

【0100】図10は、第5の実施形態において、直線予測により実難度データから予測難度データを算出する方法を例示する図である。なお、図10には、上述のように、実難度データと、実難度データから算出される予測難度データがそれぞれ15(L+A, B=15)である場合を示してある。

【0101】エンコーダ制御部12は、第1の実施形態と同様に、入力される非圧縮映像データVINに含まれるピクチャーの順番を並び替え、FIFOメモリ160およびエンコーダ162に対して出力する。FIFOメモリ160は、入力された映像データS12の各ピクチャーを15ピクチャー分(L+A=15)つまり、1GOP分だけ遅延し、エンコーダ18に対して出力する。エンコーダ162は、まず、図10中に○印を付して示すように、入力された映像データS12のピクチャーを予備的に順次、圧縮符号化し、第1番目のピクチャー~第15番目のピクチャーを圧縮符号化して得られた圧縮符号化データのデータ量、DCT処理後の映像データのDC成分の値およびAC成分の電力値をホストコンピュータ20に対して順次、出力する。ホストコンピュータ20は、これらの値に基づいて、実難度データD<sub>1</sub>~D<sub>15</sub>を順次、算出する。

【0102】ホストコンピュータ20は、算出した実難度データD<sub>1</sub>~D<sub>15</sub>の平均値D''を、下に示す式11により算出する。

【0103】

【数11】

(11)

で、傾きがaの、下に示す式13により仮の予測難度データD'<sub>1</sub>~D'<sub>15</sub>を算出し、式13により誤差関数E(a)を求める。

【0105】

$$D'_j = a \times (j - 8) + D''$$

【0106】

$$E(a) = \sum_{k=1}^{15} (D_j - D'_j)^2$$

【0107】さらに、ホストコンピュータ20は、式13に示した誤差関数E(a)を最小にする傾きaを求める。ホストコンピュータ20は、このように求めた傾きaを式12に代入し、図10中に×印で示す予測難度データD'16～D'30を算出する。以上のようにホストコンピュータ20は、予測難度データD'16～D'30を算出し、第2の実施形態および第3の実施形態に示したように、算出した予測難度データD'16～D'30を用いて目標データ量T1を算出する。

【0108】さらに、ホストコンピュータ20は、エンコーダ18が第1のピクチャーから生成する圧縮映像データのデータ量を目標データ量T1とするための量子化値Q1を算出し、エンコーダ18の量子化回路166に設定する。エンコーダ18は、ホストコンピュータ20

$$E(a) = \sum_{k=1}^{15} |D_j - D'_j|$$

【0111】なお、本実施形態の直線予測は、実難度データ数および予測難度データ数が15以外の場合にも適用できるのはいうまでもない。

#### 【0112】第6実施形態

以下、本発明の第6の実施形態として、実難度データに基づいて予測難度データを、実難度データに重み付けをして算出する方法（重み付け直線予測）を説明する。なお、同一GOPに複数の種類のピクチャータイプが混在していると、難度データの予測が複雑化する。従って、説明の便宜上、第6の実施形態においても、第5の実施形態と同様に、GOPに含まれるピクチャーが全てIピクチャーであり、実難度データと予測難度データがそれぞれ15（L+A, B=15）である場合を例に説明する。

【0113】映像データのピクチャー間の時間的な相関性は、時間間隔が長くなるほど薄れてゆくと考えられる。従って、圧縮符号化中のピクチャーの目標データ量を算出するためには、時間的に近いピクチャーの難度ほど、圧縮符号化中のピクチャーの難度と大きな相関性があると考えられる。しかしながら、第5の実施形態に示した単純な直線予測により予測難度データを算出した場合、このような、時間経過に伴う難度の相関性は考慮されない。第6の実施形態で説明する重み付け直線予測による予測難度データの算出は、圧縮符号化中のピクチャーと近いピクチャーの難度ほど誤差関数に大きな影響を与えるようにして、予測難度データの精度を向上させる

【数12】

(12)

【数13】

(13)

から設定された量子化値Q1を用いて遅延映像データS16の第1のピクチャーを圧縮符号化し、圧縮映像データVOUとして出力する。以上のように、ホストコンピュータ20は、各ピクチャーに対する目標データ量を算出し、各ピクチャーを順次、圧縮符号化する。

【0109】なお、式13には、ホストコンピュータ20が、式12に示した直線と実難度データD1～D15との2乗誤差に基づいて誤差関数E(a)を算出する場合を示したが、ホストコンピュータ20の計算量を少なくするためには、式13の代わりに、下に示す式14により、式12に示した直線と実難度データD1～D15との絶対値に基づいて誤差関数E(a)を算出してもよい。

【0110】

【数14】

(14)

ことも目的にしている。

【0114】図11は、第6の実施形態において、重み付け直線予測により実難度データから予測難度データを算出する方法を例示する図である。なお、図11には、上述のように、実難度データと、実難度データから算出される予測難度データがそれぞれ15（L+A, B=15）である場合を示してある。

【0115】エンコーダ制御部12は、第1の実施形態と同様に、入力される非圧縮映像データVINに含まれるピクチャーの順番を並び替え、FIFOメモリ160およびエンコーダ162に対して出力する。FIFOメモリ160は、入力された映像データS12の各ピクチャーを15ピクチャー分（L+A=15）つまり、1GOP分だけ遅延し、エンコーダ18に対して出力する。エンコーダ162は、まず、図11中に○印を付して示すように、入力された映像データS12のピクチャーを予備的に順次、圧縮符号化し、圧縮映像データのデータ量、DCT処理後の映像データのDC成分の値およびAC成分の電力値をホストコンピュータ20に対して出力する。ホストコンピュータ20は、これらの値に基づいて、第1番目のピクチャー～第15番目のピクチャーの圧縮符号化データの実難度データD1～D15を順次、算出する。

【0116】ホストコンピュータ20は、算出した実難度データD1～D15の平均値D''を、下に示す式15により、係数kの重み付け平均値k''を算出し、また、式



16により、実難度データ $D_1 \sim D_{15}$ それぞれに重み付け係数 $\omega$  ( $1 > \omega > 0$ ) の巾乗を乗算して重み付けし、実難度データ $D_1 \sim D_{15}$ の重み付け平均値 $D''$ を算出す

$$K'' = \frac{\sum_{k=1}^{15} \omega^{15-K} \times k}{15} \quad (15)$$

【0118】

$$D'' = \sum_{k=1}^{15} \omega^{15-K} \times D_K \quad (16)$$

【0119】つぎに、ホストコンピュータ20は、算出した係数 $k$ の重み付け平均値 $k''$ および実難度データ $D_1 \sim D_{15}$ の重み付け平均値 $D''$ を用い、下に示す式17

$$D'_j = a \times (j - k'') + D'' \quad (17)$$

【0121】さらに、式17から仮の予測難度データ $D'_1 \sim D'_{15}$ を算出し、式18により誤差関数 $E(a)$ を求める。

$$E(a) = \sum_{k=1}^{15} (D_j - D'_j)^2 \times \omega^{15-K} \quad (18)$$

【0123】さらに、ホストコンピュータ20は、式13に示した誤差関数 $E(a)$ を最小にする傾き $a$ を求める。ホストコンピュータ20は、このように求めた傾き $a$ を式12に代入し、図11中に×印で示す予測難度データ $D'_{16} \sim D'_{30}$ を算出する。以上のようにホストコンピュータ20は、予測難度データ $D'_{16} \sim D'_{30}$ を算出し、第2の実施形態および第3の実施形態に示したように、算出した予測難度データ $D'_{16} \sim D'_{30}$ を用いて目標データ量 $T_1$ を算出する。

【0124】さらに、ホストコンピュータ20は、エンコーダ18が第1のピクチャーから生成する圧縮映像データのデータ量を目標データ量 $T_1$ とするための量子化値 $Q_1$ を算出し、エンコーダ18の量子化回路166に設定する。エンコーダ18は、ホストコンピュータ20

$$E(a) = \sum_{k=1}^{15} (|D_j - D'_j| \times \omega^{15-K}) \quad (19)$$

【0127】なお、本実施形態の直線予測は、実難度データ数および予測難度数が15以外の場合にも適用できるのはいうまでもない。

#### 【0128】第7実施形態

以下、本発明の第7の実施形態として、実難度データに基づいて予測難度データを、隣接する実難度データ間の差分の変化を考慮して算出する方法（差分予測）を説明する。なお、同一GOPに複数の種類のピクチャータイプが混在していると、難度データの予測が複雑化する。従って、説明の便宜上、第7の実施形態においても、これまで説明した第5の実施形態および第6の実施形態と

る。

【0117】

【数15】

【数16】

で表される直線を定める。

【0120】

【数17】

【0122】

【数18】

から設定された量子化値 $Q_1$ を用いて遅延映像データS16の第1のピクチャーを圧縮符号化し、圧縮映像データVOUとして出力する。以上のように、ホストコンピュータ20は、各ピクチャーに対する目標データ量を算出し、各ピクチャーを順次、圧縮符号化する。

【0125】なお、式18には、ホストコンピュータ20が、式16に示した直線と実難度データ $D_1 \sim D_{15}$ との2乗誤差に基づいて誤差関数 $E(a)$ を算出する場合を示したが、ホストコンピュータ20の計算量を少なくするためには、式18の代わりに、下に示す式19により、式16に示した直線と実難度データ $D_1 \sim D_{15}$ との絶対値に基づいて誤差関数 $E(a)$ を算出してもよい。

【0126】

【数19】

同様に、GOPに含まれるピクチャーが全てIピクチャーであり、実難度データと予測難度データがそれぞれ15 ( $L+A, B=15$ ) である場合を例に説明する。

【0129】図12は、第7の実施形態において、差分予測により実難度データから予測難度データを算出する方法を例示する図である。なお、図12には、上述のように、実難度データと、実難度データから算出される予測難度データがそれぞれ15 ( $L+A, B=15$ ) である場合を示してある。

【0130】例えば、図12中に、○印で示すように、徐々に実難度データの値の増加が鈍っている絵柄の映像



データにおいては、その後（第16番目のピクチャー以降）も、映像データの難度の増加が鈍くなり、さらに、難度が減少してゆくことが予想される。このような場合、隣接する実難度データ間の値の差分を予測難度データに反映させることにより、予測難度データの精度をさらに向上させることができると考えられる。第7の実施形態に示す差分予測は、かかる観点から、予測難度データの精度を向上させようとするものである。なお、隣接する実難度データ間の値の差分を予測難度データに反映させる方法は、何種類も考えられ、いくらでも複雑化できるが、第7の実施形態においては、最も単純な、差分の値を直線近似する場合について説明する。

【0131】エンコーダ制御部12は、第1の実施形態と同様に、入力される非圧縮映像データVINに含まれるピクチャーの順番を並び替え、FIFOメモリ160およびエンコーダ162に対して出力する。FIFOメモリ160は、入力された映像データS12の各ピクチャーを15ピクチャー分（L+A=15）つまり、1GOP分だけ遅延し、エンコーダ18に対して出力する。エンコーダ162は、まず、図12中に○印を付して示すように、入力された映像データS12のピクチャーを予備的に順次、圧縮符号化し、第1番目のピクチャー～第15番目のピクチャーを圧縮符号化して得られた圧縮

$$D'_j = D'_{16} + \sum_{k=16}^j \Delta_k$$

【0135】ホストコンピュータ20は、実難度データ $D_1 \sim D_{15}$ 、および、以上説明したように算出した予測難度データ $D'_{16} \sim D'_{30}$ を用いて目標データ量 $T_1$ を算出する。なお、第7の実施形態においては、ホストコンピュータ20が隣接するピクチャー間の差分値 $\Delta_k$ を直線（1次関数）で近似する。従って、ホストコンピュータ20が算出する予測難度データ $D'_{16} \sim D'_{30}$ は、図12に×印を付して示すように1次関数を積分した放物線（2次関数）上の値に近似される。

【0136】さらに、ホストコンピュータ20は、エンコーダ18が第1のピクチャーから生成する圧縮映像データのデータ量を目標データ量 $T_1$ とするための量子化値 $Q_1$ を算出し、エンコーダ18の量子化回路166に設定する。エンコーダ18は、ホストコンピュータ20から設定された量子化値 $Q_1$ を用いて遅延映像データS16の第1のピクチャーを圧縮符号化し、圧縮映像データVOUTとして出力する。以上のように、ホストコンピュータ20は、各ピクチャーに対する目標データ量を算出し、各ピクチャーを順次、圧縮符号化する。なお、本実施形態の直線予測は、実難度データ数および予測難度数が15以外の場合にも適用できるのはいうまでもない。

#### 【0137】第8実施形態

以下、本発明の第8の実施形態として、GOPが複数の

符号化データのデータ量、DCT処理後の映像データのDC成分の値およびAC成分の電力値をホストコンピュータ20に対して出力する。ホストコンピュータ20は、これらの値から実難度データ $D_1 \sim D_{15}$ を順次、算出する。

【0132】ホストコンピュータ20は、エンコーダ162から入力された圧縮映像データのデータ量、DCT処理後の映像データのDC成分の値およびAC成分の電力値から実難度データ $D_1 \sim D_{15}$ を算出し、さらに、隣接する実難度データ間の差分値 $\Delta_k$ （ $\Delta_k = D_k - D_{k-1}$ ； $k=0, 2, \dots, 14$ ）を算出する。さらに、ホストコンピュータ20は、第5の実施形態に示した式10～式13または第6の実施形態に示した式14～式18を用いて直線近似を行い、予測差分値 $\Delta_{16} \sim \Delta_{30}$ を算出する。

【0133】さらに、ホストコンピュータ20は、算出した予測差分値 $\Delta_{16} \sim \Delta_{30}$ について、第5の実施形態に示した式11または第6の実施形態に示した式16を適用し、予測難度データ $D'_{16}$ を算出し、下に示す式20を用いて、予測難度データ $D'_{17} \sim D'_{30}$ を算出する。

【0134】

【数20】

(20)

種類のピクチャーから構成されている場合に予測難度データを算出する方法を説明する。なお、第8の実施形態においては、1GOPがI、B、B、P、B、B、P、B、B、P、B、Bの順に15のピクチャーから構成され、実難度データと予測難度データがそれぞれ15（L+A、B=15）である場合を例示する。

【0138】第5の実施形態～第7の実施形態においては、便宜上、GOPがIピクチャーのみから構成されている場合を例に本発明の説明を行った。しかしながら、実際には、GOPはIピクチャー、PピクチャーおよびBピクチャーから構成され、ピクチャータイプに対応した方法で予測難度データを算出する必要がある。第8の実施形態における予測難度データの算出方法は、かかる実情を考慮したものである。

#### 【0139】ピクチャータイプ別の予測

まず、第1にピクチャータイプ別に予測難度データを算出し、予測簡易2パスエンコードを行う方法が考えられる。つまり、エンコーダ162（図1）がGOPの各ピクチャーを圧縮符号化して得たデータ量、DCT処理後の映像データのDC成分の値およびAC成分の値をホストコンピュータ20に対して出力し、ホストコンピュータ20がこれらの値から算出した実難度データを、ホストコンピュータ20が第4の実施形態～第7の実施形態に示した単純な直線予測、重み付け直線予測または差分

予測によりピクチャタイプ別に独立して処理し、Iピクチャ、PピクチャおよびBピクチャそれぞれの予測難度データを算出し、目標データ量 $T_j$ を算出する。エンコーダ18は、第2の実施形態および第4の実施形態～第7の実施形態に示したように、ピクチャタイプ別にホストコンピュータ20が算出した目標データ量 $T_j$ に基づいて遅延映像データS16を圧縮符号化し、圧縮映像データVOU<sub>T</sub>として出力する。

#### 【0140】正規化予測

次に、GOPに含まれる各ピクチャの実難度データの値に応じて実難度データを正規化し、正規化した実難度データから予測難度データを算出し、予測簡易2パスエンコードを行う方法が考えられる。

【0141】正規化予測を行う場合、まず、ホストコンピュータ20は、エンコーダ162がGOPの各ピクチャから生成したIピクチャ、BピクチャおよびPピクチャそれぞれのデータ量、DCT処理後のDC成分の値およびAC成分の電力値から実難度データを算出し、さらに、Iピクチャ、BピクチャおよびPピクチャそれぞれの実難度データの値の比率（I：P：B）を算出する。

【0142】次に、ホストコンピュータ20は、算出した実難度データの値の比率でIピクチャ、BピクチャおよびPピクチャの実難度データを正規化し、正規化した実難度データを用いて仮の予測難度データを算出する。さらに、ホストコンピュータ20は、仮の予測難度データにIピクチャ、BピクチャおよびPピクチャそれぞれの比率を乗算して予測難度データを算出する。

【0143】以下、図13を参照して正規化予測方法をさらに具体的に説明する。図13は、第8の実施形態において、GOPが複数のタイプのピクチャから構成される場合に、実難度データから予測難度データを算出する方法を示す図である。なお、図13には、上述のように、実難度データと、実難度データから算出される予測難度データがそれぞれ15（L+A，B=15）である場合を示してある。

【0144】エンコーダ162は、映像データS12のGOPにI，B，B，P，B，B，P，B，B，P，B，B，P，B，Bの順に含まれるピクチャのデータ量、DCT処理後のDC成分の値およびAC成分の電力値をホストコンピュータ20に対して出力する。ホストコンピュータ20は、これらの値から、それぞれ図13中に○印で示す実難度データ $D_1 \sim D_{15}$ を算出する。さらに、ホストコンピュータ20は、生成した実難度データ $D_1 \sim D_{15}$ を正規化する。つまり、例えば、実難度データ $D_1 \sim D_{15}$ の内、Iピクチャの難度データ $D_1$ の値、Pピクチャの難度データ $D_4, D_7, D_{10}, D_{13}$ それぞれの値、および、Bピクチャの難度データ $D_2, D_3, D_5, D_6, D_8, D_9, D_{11}, D_{12}, D_{14}, D_{15}$ それぞれの値の比率、つまり、1つのIピクチャのデータ量、1つのPピクチャのデータ量および1つのBピクチャのデータ量の比率が、例えばほぼ4：2：1である場合に、ホストコンピュータ20は、Iピクチャの実難度データ $D_1$ を4で除算し、Pピクチャの実難度データ $D_4, D_7, D_{10}, D_{13}$ それぞれの値を2で除算し、さらに、Bピクチャの実難度データ $D_2, D_3, D_5, D_6, D_8, D_9, D_{11}, D_{12}, D_{14}, D_{15}$ それぞれの値を1で除算して正規化を行い、図13中に●印で示す正規化した実難度データを生成する。

【0145】次に、ホストコンピュータ20は、例えば、第5の実施形態に示した単純な直線予測により、図13中に×印で示す仮の予測難度データを生成する。さらに、図13中に×印で示す仮難度データの内、Iピクチャの仮難度データには4を乗算し、Pピクチャの仮難度データには2を乗算し、Bピクチャの仮難度データには1を乗算して、図13中に□印で示す予測難度データ $D'_{16} \sim D'_{30}$ を生成する。

【0146】第5の実施形態においてと同様に、ホストコンピュータ20は、実難度データ $D_1 \sim D_{15}$ および予測難度データ $D'_{16} \sim D'_{30}$ を用いて目標データ量 $T_j$ を生成する。エンコーダ18は、ホストコンピュータ20が生成した目標データ量 $T_j$ に基づいて、遅延映像データS16を圧縮符号化し、圧縮映像データVOU<sub>T</sub>として出力する。

【0147】第9実施形態

以下、本発明の第9の実施形態を説明する。第1の実施形態～第8の実施形態に示した映像データ圧縮装置1によれば、短時間で非圧縮映像データVINの絵柄の難度を算出し、算出した難度に応じた圧縮率で適応的に非圧縮映像データVINを圧縮符号化することができる。しかしながら、第1の実施形態において説明したように、エンコーダ162の量子化回路168（図1，図2）に設定される量子化値Qの値は、非圧縮映像データVINの絵柄の複雑さ、あるいは動き等にかかわらず一定である。

#### 【0147】第9実施形態

【0148】例えば、第1の実施形態において、表1および表2を参照して説明したように、エンコーダ162が生成する圧縮映像データ15ピクチャ分のデータ量（難度データ）が1744380ビットであり、15ピクチャの圧縮映像データに割り当てるデータ量が3000000であるということは、エンコーダ18の量子化回路168で実際に用いられている量子化値と、エンコーダ162に設定されている固定値の量子化値Qとの間に、大きな差があるということを示している。つまり、エンコーダ162が生成する圧縮映像データのデータ量が、エンコーダ18が生成する圧縮映像データに割り当てられるデータ量になるべく近づくようにエンコーダ18の量子化回路168に設定する量子化値Qの値を

制御することにより、より精度よく非圧縮映像データVINの難度を検出することができる。具体的には、表1および表2に示した例においては、エンコーダ162が生成する圧縮映像データ15ピクチャー分のデータ量（難度データ）が3000000ビットに近ければ近いほど、ホストコンピュータ20は、誤差が少ない目標データ量 $T_j$ を算出することができる。

【0149】第9の実施形態に示す映像データ圧縮装置2は、以上説明した第1～第8の実施形態に示した映像データ圧縮装置1の問題点を解決するためになされたものであり、映像データの時間方向の相関性に着目し、予測系に設定する量子化値Qの値に対して、本線系の量子化回路に設定する量子化値に応じて調節するフィードバック制御を行い、入力映像データVINの難度および目標データ量 $T_j$ を精度高く算出し、より高品質な圧縮映像データVOUTを生成可能なように構成されている。

【0150】図14は、第9の実施形態における本発明に係る映像データ圧縮装置2の構成を示す図である。なお、図14において、映像データ圧縮装置2の構成部分の内、第1～第8の実施形態において説明した映像データ圧縮装置1の構成部分（図1～図3）と同じものには、同一の符号を付してある。図14に示すように、映像データ圧縮装置2は、予測系22、エンコーダ18、ホストコンピュータ20および量子化値生成部（Q生成）24から構成され、予測系22は、エンコーダ制御部12、モーションエスティメータ14、簡易2パス処理部16、FIFOメモリ186およびバッファ回路182、184から構成される。

【0151】つまり、映像データ圧縮装置2は、映像データ圧縮装置1に、量子化値生成部24、FIFOメモリ186およびバッファ回路182、184を付加した構成を採る。なお、図14において、エンコーダ制御部12は、画像並び替え部12.0と走査変換マクロブロック化部12.2とに分割されているが、その動作は第1～第8の実施形態におけるエンコーダ制御部12と同一である。

【0152】映像データ圧縮装置2において、バッファ回路182は、エンコーダ162が生成した圧縮映像データのデータ量、DC値およびAC電力をバッファリングして、ホストコンピュータ20に対して出力する。映像データ圧縮装置2において、バッファ回路184は、エンコーダ18が生成した圧縮映像データをバッファリングし、圧縮映像データVOUTとして外部に出力する。FIFOメモリ186は、モーションエスティメータ14が生成した動きベクトルをバッファリングして、エンコーダ18の動き補償回路178に対して出力する。

【0153】以下、簡易2パス処理部16のエンコーダ162が生成する圧縮映像データのデータ量を、そのまま難度データとして用いる場合を例に、映像データ圧縮

装置2の量子化値生成部24の動作を説明する。量子化値生成部24は、エンコーダ18の量子化制御回路180が生成し、量子化回路168に設定する量子化値 $R_j$ に基づいて、エンコーダ162が生成する圧縮映像データのデータ量が、エンコーダ18が生成する圧縮映像データに元々、割り当てられているデータ量に近づくような量子化値 $R_j'$ をピクチャーごとに生成し、エンコーダ162の量子化回路168に設定する。

【0154】なお、第9の実施形態においては、量子化値生成部24は、例えば、エンコーダ18が第n番目のピクチャーを圧縮符号化する間に、量子化制御回路180が量子化回路168に設定した量子化値 $R_j$ の平均値を、エンコーダ162が第(n+L+1)番目[LはFIFOメモリ160の遅延時間（ピクチャー数）]のピクチャーを圧縮符号化する際に用いる量子化値 $R_j'$ として算出する。

【0155】以下、第9の実施形態における映像データ圧縮装置2の動作を説明する。非圧縮映像データVINのピクチャーが、第1番目から第L+1番目まで入力される間、エンコーダ18が量子化回路168に設定する量子化値 $R_j$ が1ピクチャー分そろわないため、量子化値生成部24は、量子化値 $R_j'$ の初期値をエンコーダ162の量子化回路168に設定する。エンコーダ162の量子化回路168は、設定された量子化値 $R_j'$ の初期値を用いて圧縮符号化を行う。

【0156】ホストコンピュータ20は、第1～第8の実施形態に示したように、エンコーダ162の圧縮映像データのデータ量に基づいて目標データ量 $T_j$ を生成し、エンコーダ18の量子化回路168に設定する。エンコーダ18は、ホストコンピュータ20により設定された目標データ量 $T_j$ が示すデータ量にほぼ等しいデータ量の圧縮映像データVOUTを生成し、外部に出力する。

【0157】非圧縮映像データVINの第(L+1)番目のピクチャーが入力されると、エンコーダ18は、非圧縮映像データVINに対する圧縮符号化を開始し、エンコーダ18の量子化制御回路180は、量子化値 $R_{L+1}$ の生成、および、量子化回路168に対する設定を開始する。

【0158】量子化値生成部24は、非圧縮映像データVINの第(L+1)番目のピクチャーに対する圧縮符号化処理の際に、エンコーダ18の量子化制御回路180が出力する量子化値 $R_{L+1}$ を順次、取り込み、平均値を算出して、エンコーダ162に対する量子化値 $R_{L+1}'$ を生成する。エンコーダ18が非圧縮映像データVINの第(L+1)番目のピクチャーに対する圧縮符号化処理をしている間の量子化値 $R_{L+1}$ に基づいて算出された量子化値 $R_{L+1}'$ は、エンコーダ162が非圧縮映像データVINの第(2L+1)番目のピクチャーに対する圧縮符号化処理を行う際に、エンコーダ162

の量子化回路168に設定され、量子化回路168は、設定された量子化値 $R_{L+1}$ を用いて、非圧縮映像データVINの第 $(2L+1)$ 番目のピクチャーに対する量子化処理を行う。

【0159】以下、同様に、量子化値生成部24は、量子化値 $R_j$ の生成およびエンコーダ162の量子化回路168に対する設定を行う。つまり、量子化値生成部24は、エンコーダ18が非圧縮映像データVINの $n$ 番目 $(n \geq L+1)$ のピクチャーに対して圧縮符号化処理をする間に、エンコーダ18の量子化制御回路180が出力する量子化値 $R_n$ を順次、取り込み、これらの量子化値 $R_n$ の平均値を算出して、エンコーダ162に対する量子化値 $R_n$ を生成する。

【0160】さらに、量子化値生成部24は、量子化値 $R_n$ を、エンコーダ162が非圧縮映像データVINの第 $(n+L+1)$ 番目のピクチャーに対する圧縮符号化処理に用いる量子化値 $R_{n+L+1}$ として、エンコーダ162の量子化回路168に設定する。エンコーダ162の量子化回路168は、量子化値生成部24から設定された量子化値 $R_{n+L+1}$ に基づいて量子化処理を行い、圧縮映像データを生成する。エンコーダ162は、量子化した映像データを可変長符号化し、バッファ回路182を介してホストコンピュータ20に対して出力する。ホストコンピュータ20は、エンコーダ162が生成した圧縮映像データのデータ量に基づいて、目標データ量 $T_j$ を生成し、エンコーダ18に設定する。エンコーダ18は、ホストコンピュータ20から設定された目標データ量 $T_j$ に基づいて、非圧縮映像データVINの第 $(n+1)$ 番目のピクチャーに対する圧縮符号化処理を行う。

【0161】以下、図15を参照して、第9の実施形態における量子化値生成部24の動作をさらに説明する。

図15は、第9の実施形態における量子化値生成部24(図14)の量子化値 $R_j$ の算出動作を示すフローチャートである。図15に示すように、ステップ300

(S300)において、エンコーダ18が非圧縮映像データVINの第 $n$ 番目のピクチャーに対する圧縮符号化を開始すると、まず、量子化値生成部24は、数値1、sumの値を0にする(ゼロクリア)する。ステップ302(S302)において、量子化値生成部24は、エンコーダ18の量子化制御回路180が、量子化回路168に設定する量子化値 $R_j$ の総数 $(N+1)$ を示す数値 $N$ と、数値1とを比較し、両者が等しい場合にはS308の処理に進み、数値1が数値 $N$ より小さい場合には、S304の処理に進む。

【0162】ステップ304(S304)において、量子化値生成部24は、エンコーダ18の量子化制御回路180が出力する量子化値 $(Q)R_j$ を、数値sumに加算する。ステップ306(S306)において、量子化値生成部24は、数値1に1を加算(数値1をインク

リメント)する。ステップ308(S308)において、量子化値生成部24は、数値sumを数値 $N$ で除算し、平均値(Qave)を算出し、量子化値 $R_n$ としてエンコーダ162の量子化回路168に設定する。

【0163】なお、上述したように、非圧縮映像データVINの第 $n$ 番目のピクチャーから生成され、非圧縮映像データVINの第 $(n+L+1)$ 番目のピクチャーの圧縮符号化のためにエンコーダ162の量子化回路168に設定される量子化値 $R_n$ は、エンコーダ162が実際に圧縮符号化するピクチャーに比べて、ピクチャー $L$ 枚分の差があるピクチャーから生成されるが、映像データは時間方向に対して高い相関性を有するので、量子化値 $R_j$ を適応制御しない場合に比べて、エンコーダ18が生成する圧縮映像データVOU Tの品質が向上する。

【0164】また、量子化値生成部24が、非圧縮映像データVINの複数のピクチャーを圧縮符号化する際の量子化値 $R_j$ の平均値を算出することにより、つまり例えば、量子化値生成部24が、非圧縮映像データの第 $n-L$ 番目～第 $n$ 番目のピクチャーを圧縮符号化する際の量子化値 $R_j$ の平均値を、量子化値 $R_j$ として算出することにより、エンコーダ18が生成する圧縮映像データVOU Tの品質がさらに向上する。また、量子化値生成部24が、非圧縮映像データVINの複数のピクチャーを圧縮符号化する際の量子化値 $R_j$ の平均値を算出することにより、さらに最適な量子化値 $R_j$ を算出することができ、エンコーダ18が生成する圧縮映像データVOU Tの品質がさらに向上する。

【0165】また、第9の実施形態に示した量子化値 $R_j$ の適応制御と、第1～第8の実施形態に示した映像データ圧縮装置1の各動作とを、矛盾を生じない限りにおいて、組み合わせる用いることが可能である。また、第9の実施形態においては、量子化値 $R_j$ の平均値を量子化値 $R_j$ として用いる場合について説明したが、例えば、新しい量子化値 $R_j$ ほど量子化値 $R_j$ に対して、大きな影響を与えるように重み付けするように量子化値生成部24の動作を変形してもよい。また、動きベクトル等、他のデータを考慮して、量子化値 $R_j$ から量子化値 $R_j$ を算出するように量子化値生成部24の動作を変形してもよい。

#### 【0166】第10実施形態

以下、本発明の第10の実施形態を説明する。第9の実施形態においては、量子化値生成部24(図14)が、圧縮映像データのピクチャータイプを考慮せずに、エンコーダ18の量子化制御回路180が生成する量子化値 $R_j$ から、エンコーダ162の量子化回路168に対する量子化値 $R_j$ を生成する場合の映像データ圧縮装置2の動作を説明した。しかしながら、圧縮映像データのデータ量は、ピクチャータイプごとに大きく異なるので、量子化値 $R_j$ は、Iピクチャー、Bピクチャーお

およびPピクチャーごとに生成するのが適当である。第10の実施形態には、ピクチャータイプシーケンスを周期として、量子化値 $R_{ji}'$ をIピクチャー、BピクチャーおよびPピクチャー対応に生成する映像データ圧縮装置2(図14)の動作を説明する。

【0167】以下、簡易2パス処理部16のエンコーダ162が生成する圧縮映像データのデータ量を、そのまま難度データとして用いる場合を例に、第10の実施形態における映像データ圧縮装置2の量子化値生成部24の動作を説明する。量子化値生成部24は、エンコーダ18が圧縮符号化後にそれぞれIピクチャー、BピクチャーおよびPピクチャーになる非圧縮映像データVINを圧縮符号化する際に、量子化制御回路180が生成する量子化値 $R_{ji}$ 、 $R_{jb}$ 、 $R_{jp}$ それぞれに基づいて、エンコーダ162が生成する圧縮映像データのデータ量が、エンコーダ18が生成する圧縮映像データに元々、割り当てられているデータ量に近づくような量子化値 $R_{ji}'$ 、 $R_{jb}'$ 、 $R_{jp}'$ を、各ピクチャータイプのピクチャーごとに生成し、エンコーダ162の量子化回路168に設定する。

【0168】なお、第10の実施形態においては、量子化値生成部24は、第9の実施形態においてと同様に、例えば、エンコーダ18が圧縮符号化した後にIピクチャー、BピクチャーおよびPピクチャーとなる非圧縮映像データのピクチャーを圧縮符号化する際に、量子化制御回路180が量子化回路168に設定した量子化値 $R_{ji}$ 、 $R_{jb}$ 、 $R_{jp}$ の平均値を、それぞれ、エンコーダ162が圧縮符号化した後にIピクチャー、BピクチャーおよびPピクチャーとなる非圧縮映像データのピクチャーを圧縮符号化する際に用いる量子化値 $R_{ji}'$ 、 $R_{jb}'$ 、 $R_{jp}'$ として算出する。また、量子化値 $R_{ji}'$ 、 $R_{jb}'$ 、 $R_{jp}'$ の算出は、例えば、エンコーダ18およびエンコーダ162が生成する圧縮映像データのピクチャータイプシーケンスを周期として行われる。

【0169】以下、第10の実施形態における映像データ圧縮装置2の動作を説明する。映像データ圧縮装置2に、第(L+1)番目の非圧縮映像データVINのピクチャーが入力されて以降、エンコーダ18は、順次、非圧縮映像データVINに対する圧縮符号化を開始し、エンコーダ18の量子化制御回路180は、量子化値 $R_{ji}$ 、 $R_{jb}$ 、 $R_{jp}$ の生成、および、量子化回路168に対する設定を開始する。

【0170】量子化値生成部24は、エンコーダ18の量子化制御回路180が出力する量子化値 $R_j$ を順次、取り込み、ピクチャータイプシーケンスを周期とし、ピクチャータイプごとに平均値を算出して、エンコーダ162に対する量子化値 $R_{ji}'$ 、 $R_{jb}'$ 、 $R_{jp}'$ を生成する。エンコーダ162の量子化回路168は、量子化値生成部24が量子化値 $R_{ji}'$ 、 $R_{jb}'$ 、 $R_{jp}'$ をそれぞれ生成した直後に入力され、圧縮符号化後にIピクチャー

一、BピクチャーおよびPピクチャーになる非圧縮映像データVINのピクチャーを、量子化値 $R_{ji}'$ 、 $R_{jb}'$ 、 $R_{jp}'$ を用いて量子化し、可変長符号化回路170およびバッファ回路182を介してホストコンピュータ20に対して出力する。

【0171】ホストコンピュータ20は、エンコーダ162が生成した圧縮映像データのデータ量に基づいて、次のピクチャータイプシーケンスに含まれる非圧縮映像データVINを圧縮符号化する際の目標データ量 $T_j$ を生成し、エンコーダ18に設定する。エンコーダ18は、ホストコンピュータ20から設定された目標データ量 $T_j$ に基づいて、非圧縮映像データVINの各ピクチャーに対する圧縮符号化処理を行う。

【0172】以下、図16を参照して、第10の実施形態における量子化値生成部24の処理をさらに説明する。図16は、第10の実施形態における量子化値生成部24(図14)の量子化値 $R_j'$ の算出動作を示すフローチャートである。図16に示すように、ステップ400(S400)において、量子化値生成部24は、数値 $l$ 、 $N\_i$ 、 $N\_p$ 、 $N\_b$ 、 $sum\_i$ 、 $sum\_p$ 、 $sum\_b$ の値を0にする(ゼロクリア)する。なお、数値 $N\_i$ 、 $N\_p$ 、 $N\_b$ はそれぞれ、エンコーダ18が、Iピクチャー、PピクチャーおよびBピクチャーを生成する際の量子化値 $R_{ji}$ 、 $R_{jp}$ 、 $R_{jb}$ の数を示し、数値 $sum\_i$ 、 $sum\_p$ 、 $sum\_b$ はそれぞれ、エンコーダ18が、Iピクチャー、PピクチャーおよびBピクチャーを生成する際の量子化値 $R_{ji}$ 、 $R_{jp}$ 、 $R_{jb}$ の総和値を示す。

【0173】ステップ402(S402)において、量子化値生成部24は、エンコーダ18の量子化制御回路180が、1組のピクチャータイプシーケンスに含まれるIピクチャー、PピクチャーおよびBピクチャーを圧縮符号化する際に量子化回路168に設定する量子化値 $R_j$ の総数( $N+1=N\_i+N\_p+N\_b$ )を示す数値 $N$ と、数値 $l$ とを比較し、両者が等しい場合にはS420の処理に進み、数値 $l$ が数値 $N$ より小さい場合には、S404の処理に進む。

【0174】ステップ404(S404)において、量子化値生成部24は、エンコーダ18が、いずれのピクチャーとなる非圧縮映像データVINのピクチャーを圧縮符号化しているかを判断する。エンコーダ18がそれぞれ、Bピクチャー、PピクチャーおよびIピクチャーとなる圧縮映像データVINを圧縮符号化している場合には、量子化値生成部24はそれぞれ、S406、S410、S414の処理に進む。

【0175】ステップ406(S406)において、量子化値生成部24は、数値 $sum\_b$ に量子化値(Q) $R_{jb}$ を加算する。ステップ408(S408)において、量子化値生成部24は、数値 $N\_b$ に1を加算(インクリメント)する。

【0176】ステップ410 (S410)において、量子化値生成部24は、数値 $sum\_p$ に量子化値(Q) $R_{jp}$ を加算する。ステップ412 (S412)において、量子化値生成部24は、数値 $N\_p$ をインクリメントする。

【0177】ステップ414 (S414)において、量子化値生成部24は、数値 $sum\_i$ に量子化値(Q) $R_{ji}$ を加算する。ステップ416 (S416)において、量子化値生成部24は、数値 $N\_i$ をインクリメントする。ステップ418 (S418)において、量子化値生成部24は、数値1をインクリメントする。

【0178】ステップ420 (S420)において、量子化値生成部24は、平均値 $Qave\_i (=sum\_i / N\_i)$ 、 $Qave\_p (=sum\_p / N\_p)$ 、 $Qave\_b (=sum\_b / N\_b)$ を算出し、エンコーダ162が、それぞれ圧縮符号化後にIピクチャー、PピクチャーおよびBピクチャーとなる非圧縮映像データのピクチャーを圧縮符号化する際に、量子化回路168に設定する。

【0179】なお、量子化値 $R_{ji}$ 、 $R_{jp}$ 、 $R_{jb}$ の算出は、必ずしもピクチャータイプシーケンスを周期として行う必要はなく、任意の周期で行うことができる。また、第10の実施形態に示した映像データ圧縮装置2の動作に対しても、第9の実施形態に示した変形が可能である。また、第10の実施形態に示した映像データ圧縮装置2の動作によれば、第9の実施形態に示した映像データ圧縮装置2の動作に比べて、各ピクチャータイプに対応した、より適切な量子化値の制御が可能であり、この結果、エンコーダ18が生成する圧縮映像データVOUTの品質が向上する。

【0180】

【発明の効果】以上説明したように、本発明に係る映像データ圧縮装置およびその方法によれば、2パスエンコードによらずに所定のデータ量以下に圧縮符号化した圧縮映像データを伸長復号した場合に、高品質な映像を得ることができる。また、本発明に係る映像データ圧縮装置およびその方法によれば、2パスエンコードによらずに、圧縮符号化後の圧縮映像データのデータ量を見積もり、見積もった圧縮映像データのデータ量に基づいて圧縮率を調節して非圧縮映像データを圧縮符号化することができる。また、本発明に係る映像データ圧縮装置およびその方法によれば、2パスエンコードによらずに、高精度に圧縮符号化後の圧縮映像データのデータ量を見積もりながら、非圧縮映像データを圧縮符号化することができる。

【図面の簡単な説明】

【図1】本発明に係る映像データ圧縮装置の構成を示す図である。

【図2】図1に示した簡易2パス処理部のエンコーダの構成を示す図である。

【図3】図1に示したエンコーダの構成を示す図である。

【図4】(A)～(C)は、第1の実施形態における映像データ圧縮装置の簡易2パスエンコードの動作を示す図である。

【図5】(A)～(C)は、第2の実施形態における映像データ圧縮装置の予測簡易2パスエンコードの動作を示す図である。

【図6】第2の実施形態における映像データ圧縮装置(図1)の動作を示すフローチャートである。

【図7】(A)～(C)は、第3の実施形態において、本発明に係る映像データ圧縮装置1がGOPの枠内で行う簡易2パスエンコード方式を示す図である。

【図8】(A)～(C)は、第4の実施形態において、本発明に係る映像データ圧縮装置がGOPの枠内で行う予測簡易2パスエンコード方式を示す図である。

【図9】第4の実施形態における映像データ圧縮装置の動作を示すフローチャートである。

【図10】第5の実施形態において、直線予測により実難度データから予測難度データを算出する方法を例示する図である。

【図11】第6の実施形態において、重み付け直線予測により実難度データから予測難度データを算出する方法を例示する図である。

【図12】第7の実施形態において、差分予測により実難度データから予測難度データを算出する方法を例示する図である。

【図13】第8の実施形態において、GOPが複数のタイプのピクチャーから構成される場合に、実難度データから予測難度データを算出する方法を示す図である。

【図14】第9の実施形態における本発明に係る映像データ圧縮装置の構成を示す図である。

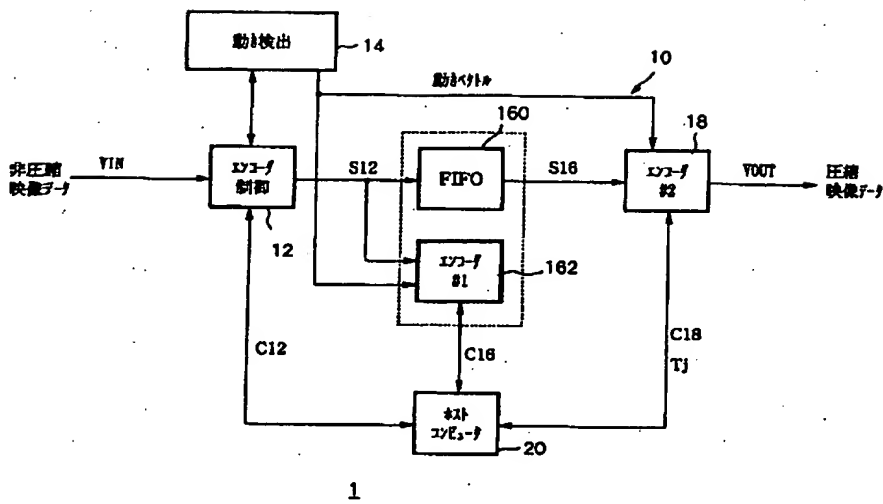
【図15】第9の実施形態における量子化値生成部(図14)の量子化値 $R_j$ の算出動作を示すフローチャートである。

【図16】第10の実施形態における量子化値生成部(図14)の量子化値 $R_j$ の算出動作を示すフローチャートである。

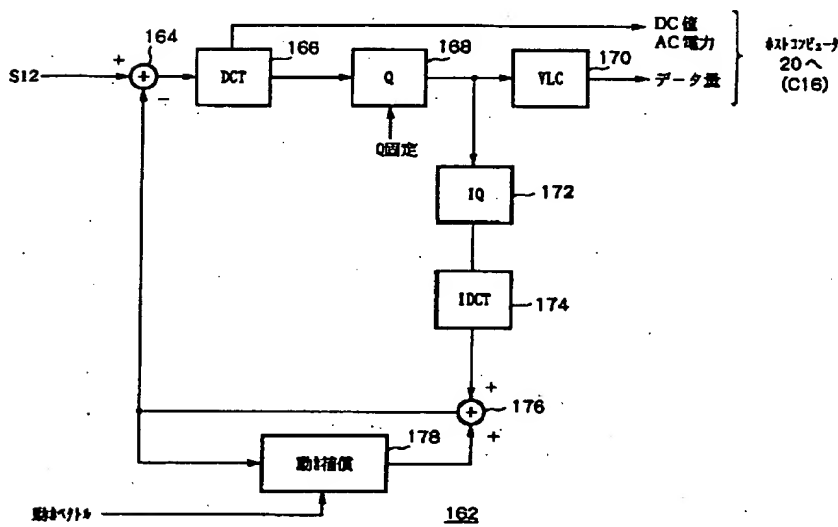
【符号の説明】

1…映像データ圧縮装置、10…圧縮符号化部、12…エンコーダ制御部、120…画像並び替え部、122…走査変換マクロブロック化部、14…動き検出器、16…簡易2パス処理部、160…FIFOメモリ、162…簡易2パス処理部、164…加算回路、166…DC T回路、168…量子化回路、170…可変長符号化回路、178…動き補償回路、180…量子化制御回路、182、184…バッファ回路、184…FIFOメモリ、160…FIFOメモリ、162…エンコーダ(予測系)、18…エンコーダ(本線系)、20…ホストコンピュータ、22…予測系。

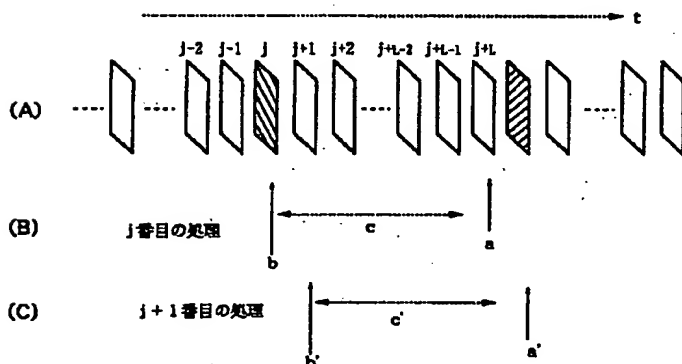
【図1】



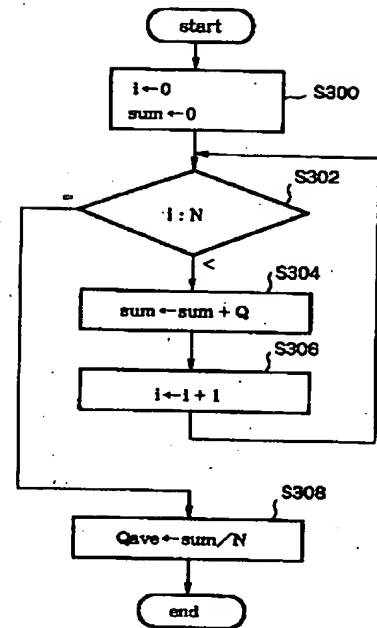
【図2】



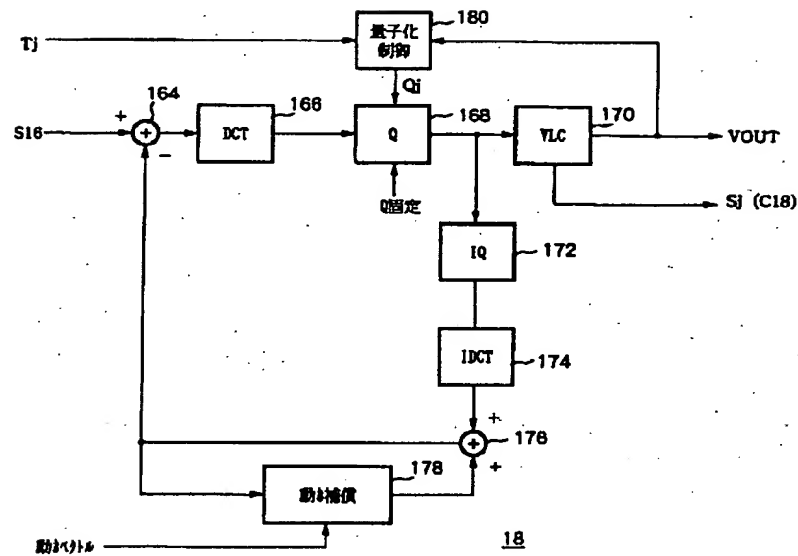
【図4】



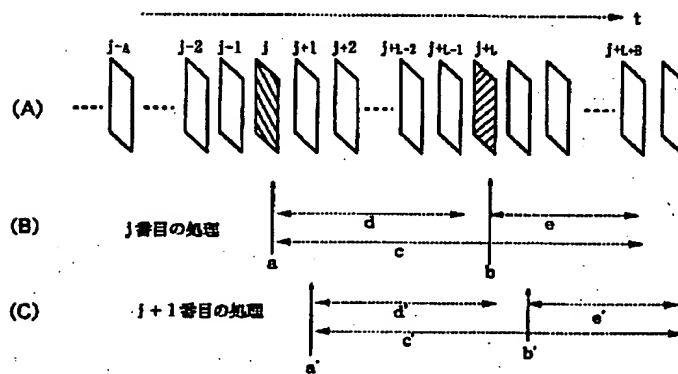
【図15】



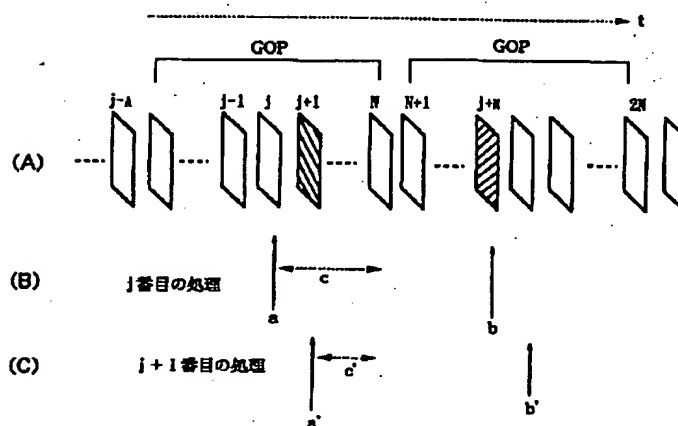
【図3】



【図5】

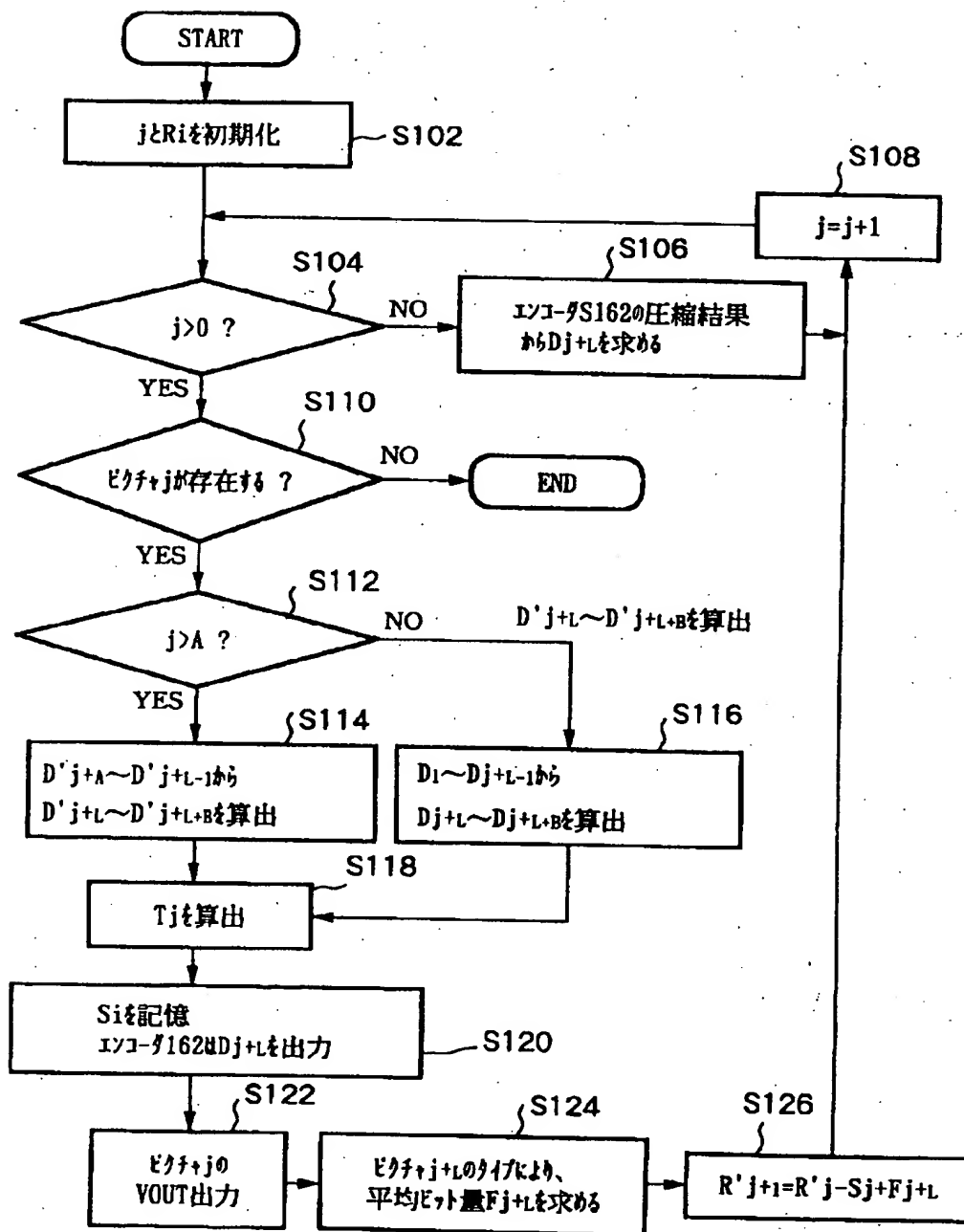


【図7】

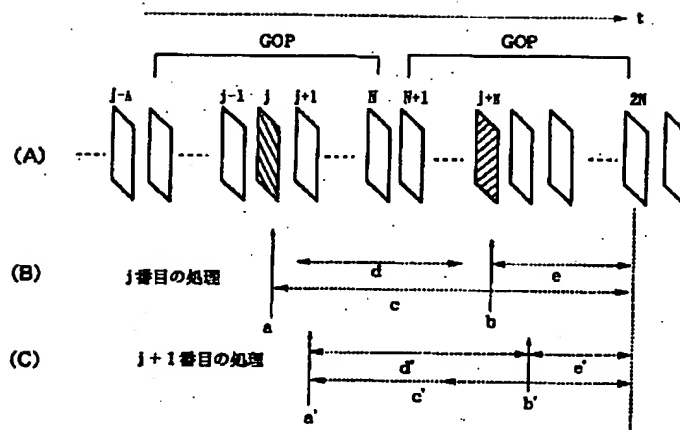




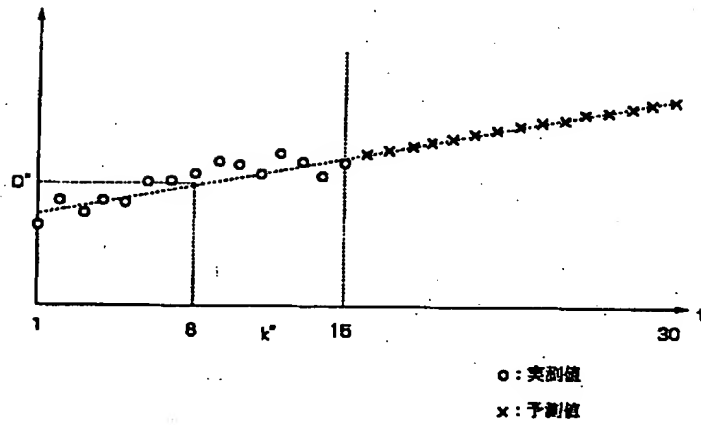
【図6】



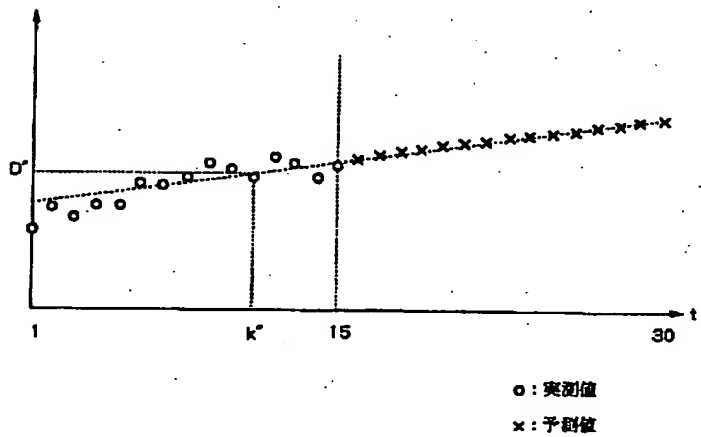
【図8】



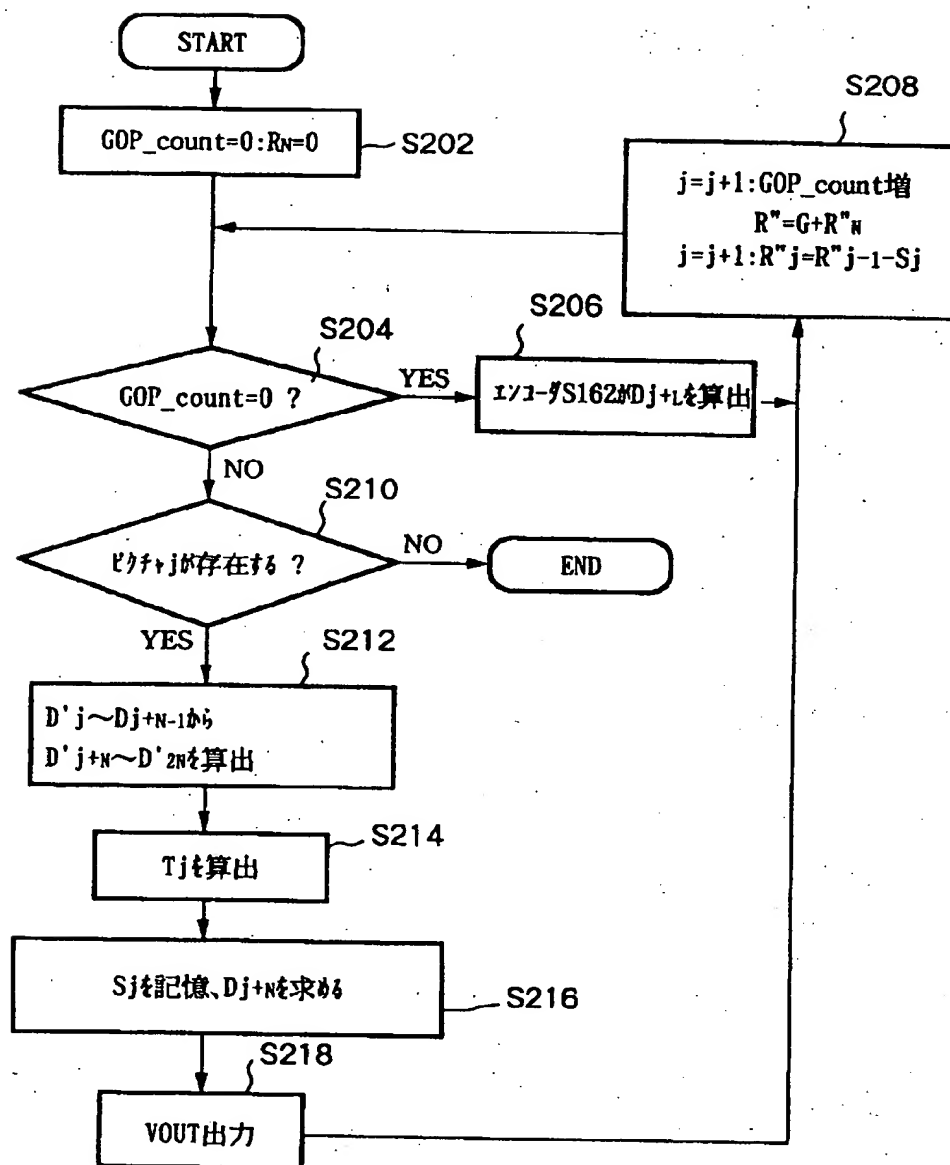
【図10】



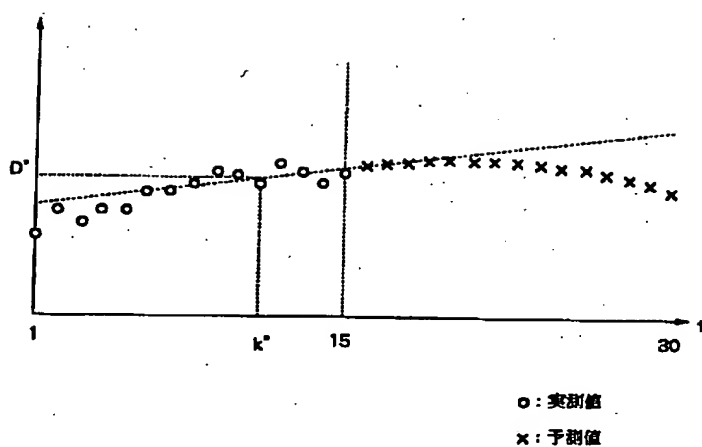
【図11】



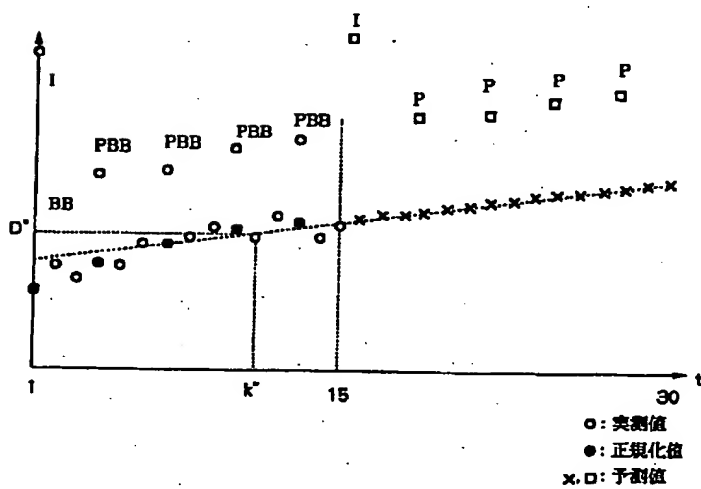
【図9】



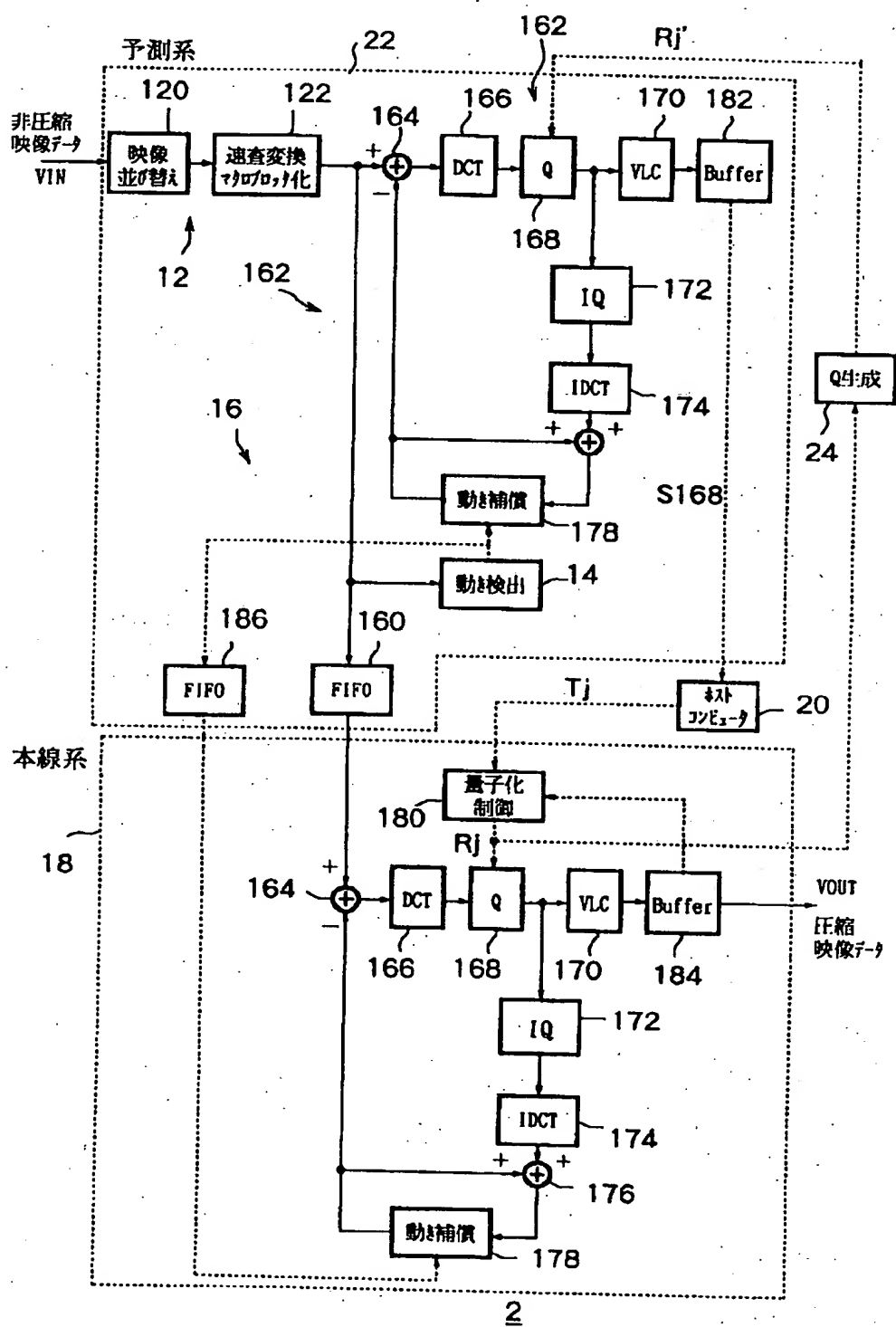
【図12】



【図13】



【図 14】



【図16】

